

ΘΕΜΑ 2

2.1. Να ξαναγράψετε το παρακάτω τμήμα προγράμματος, χρησιμοποιώντας την εντολή επανάληψης **for** αντί της εντολής επανάληψης **while** έτσι ώστε να εμφανίζει το ίδιο αποτέλεσμα.

```
1    s=0
2    x=1
3    while x<10:
4        s=s+x
5        x=x+2
6    print s
```

Μονάδες 6

2.2 Να συμπληρώσετε στο γραπτό σας την στήλη «**Αποτέλεσμα στην οθόνη**» όπου θα γράφετε το γράμμα **α, β, γ, δ** και **ε** και μετά το αποτέλεσμα που θα εμφανιστεί στο περιβάλλον της γλώσσας Python μετά την εκτέλεση της/των εντολής/εντολών της αντίστοιχης στήλης «**Εντολές**».

Εντολές	Αποτέλεσμα στην Οθόνη
α. x=2 print "x=",x	α.
β. a,b,c=2.7,3,"Python" print c,a	β.
γ. y=5 z=3 print y/z, z%y	γ.
δ. e=7 f=2 print e>f	δ.
ε. g="Arcadia" print g*3	ε.

Μονάδες 9

2.3 Το παρακάτω τμήμα προγράμματος Python υπολογίζει το άθροισμα των αριθμών 10+20+30+40+50:

```
1    sum=_(1)
2    for i in range (_(2) ,_(3) ,_(4) ):
3        sum=sum+_(5)
4    print "Το αποτέλεσμα είναι " , sum
```

Στο τμήμα προγράμματος υπάρχουν πέντε (5) κενά, τα οποία έχουν αριθμηθεί και υπογραμμιστεί. Να γράψετε στο γραπτό σας τον αριθμό του κενού και δίπλα τι πρέπει να συμπληρωθεί, ώστε το τμήμα προγράμματος να εκτελεί σωστά τη λειτουργία του.

Μονάδες 10

ΘΕΜΑ 2

2.1

```
1      s=0
2      for x in range (1,10,2):
3          s=s+x
4      print s
```

2.2

Αποτέλεσμα στην Οθόνη
α. x=2
β. Python 2.7
γ. 1 3
δ. True
ε. ArcadiaArcadiaArcadia

2.3

1. 0
2. 10
3. 51
4. 10
5. sum

ΘΕΜΑ 4

Ένας Δήμος στο πρόγραμμα των Χριστουγεννιάτικων Εκδηλώσεων διεξάγει ένα διαγωνισμό καλλιτεχνικού πατινάζ στο παγοδρόμιο που έχει στηθεί στην κεντρική πλατεία με την συμμετοχή των μαθητών των σχολείων της πόλης. Κάθε μαθητής που συμμετέχει λαμβάνει μια τελική βαθμολογία από 1 έως και 100 ακέραιες μονάδες από μία κριτική επιτροπή. Να γράψετε πρόγραμμα σε γλώσσα προγραμματισμού Python, το οποίο:

4.1. Να διαβάζει το ονοματεπώνυμο κάθε μαθητή τη βαθμολογία που έλαβε και το όνομα του σχολείου του. Τα στοιχεία αυτά καταχωρίζονται στις λίστες ON, VATH και SCH αντίστοιχα. Να γίνεται έλεγχος ορθότητας των τιμών της βαθμολογία ώστε αυτή να είναι από 1 έως και 100. Η είσοδος των στοιχείων σταματάει όταν δοθεί ως όνομα σχολείου η λέξη “STOP”.

Μονάδες 10

4.2. Να υπολογίζει και να εμφανίζει τον μέσο όρο της βαθμολογίας όλων των μαθητών.

Μονάδες 6

4.3. Να εντοπίζει και να εμφανίζει με κατάλληλο μήνυμα, το όνομα, τη βαθμολογία και το όνομα του σχολείου, του μαθητή που συγκέντρωσε την υψηλότερη βαθμολογία.

Μονάδες 9

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```


ΘΕΜΑ 4

```
1  ON=[]    #δημιουργία κενής λίστας
2  VATH=[]  #δημιουργία κενής λίστας
3  SCH=[]   #δημιουργία κενής λίστας
4  #4.1
5  name_sch=raw_input('Δώσε όνομα σχολείου μαθητή')
6  while name_sch!='STOP':    #μέχρι να δοθεί ως όνομα η τιμή STOP επανάλαβε
7      name_math=raw_input('Δώσε ονοματεπώνυμο μαθητή')
8      ON.append(name_math)    #εισαγωγή στη λίστα ON
9      bath=int(input('Δώσε βαθμολογία μαθητή')) #έναρξη ελέγχου αποδεκτών τιμών
10     while (bath<1) or (bath>100):    #έλεγχος αποδεκτών τιμών για τιμές
11         print "Έδωσες λάθος βαθμολογία"    # από 1 έως 100
12         bath=int(input('Δώσε νέα αποδεκτή βαθμολογία μαθητή'))
13     VATH.append(bath)    #εισαγωγή αποδεκτού βαθμού στη λίστα VATH
14     SCH.append(name_sch) #εισαγωγή το όνομα σχολείου στη λίστα SCH
15     name_sch=raw_input('Δώσε επόμενο όνομα σχολείου μαθητή ή STOP για τέλος')
16 #4.2 Υπολογισμός Αθροίσματος και Μέσου Όρου της λίστας VATH
17 sum=0.0    #αρχικοποίηση αθροιστή
18 pl=len(VATH)    #μήκος λίστας (πλήθος στοιχείων λίστας)
19 print pl
20 for i in range (pl):
21     sum=sum+VATH[i] #υπολογισμός αθροίσματος βαθμών
22 mo=sum/pl    #υπολογισμός μέσου όρου βαθμολογιών
23 print 'Ο μέσος όρος της βαθμολογίας των μαθητών είναι',mo
24 #4.3 Υπολογισμός μέγιστου βαθμού και εύρεση ονόματος μαθητή και σχολείου
25 meg=VATH[0]    #αρχικοποίηση μεταβλητής μεγίστου
26 on_math=ON[0]
27 on_sch=SCH[0]
28 for i in range (pl):
29     if VATH[i]>meg:
30         meg=VATH[i]
31         on_math=ON[i]
32         on_sch=SCH[i]
33 print 'Ο μαθητής',on_math,'έχει την μεγαλύτερη βαθμολογία η οποία είναι',meg
34 print 'και πηγαίνει στο σχολείο',on_sch
```


ΘΕΜΑ 2

2.1. Να γράψετε στο γραπτό σας τι εμφανίζει το παρακάτω πρόγραμμα κατά την εκτέλεση του.

```
1 L=[4, 12.0, 5, 'Python', 7, -3, '3.2', 6, 0, 'True']
2 for i in L:
3     print i
```

Μονάδες 10

2.2. Δίνεται το παρακάτω τμήμα προγράμματος :

```
1 #Ελεγχος αν ένας αριθμός είναι μονός ή ζυγός
2 ar=int(input('Δώσε έναν ακέραιο αριθμό'))
3 import math
4 if ar<0:
5     ar=abs(ar)
6 if ar%2==0:
7     print 'Ο αριθμός ', ar , ' είναι ζυγός'
8 else:
9     print 'Ο αριθμός ', ar , ' είναι μονός'
10 tet=ar**2
11 print 'Το τετράγωνο του ',ar , ' είναι ', tet
12 riza=math.sqrt(ar)
13 print 'Η τετραγωνική ρίζα του αριθμού ', ar , ' είναι ',riza
```

Να γράψετε στο γραπτό σας:

α) τις μεταβλητές,

β) Τους αριθμητικούς τελεστές,

γ) τις ενσωματωμένες συναρτήσεις,

δ) τις εξωτερικές συναρτήσεις,

ε) τις λογικές συνθήκες,

που υπάρχουν στο παραπάνω πρόγραμμα της Python.

Μονάδες 5

2.3. Να χαρακτηρίσετε καθεμιά από τις ακόλουθες λογικές εκφράσεις ως **True** ή **False** αν $a=4$ και $b=-2$:

α. $a>4$ and $b\leq 1$

β. $a\leq 5$ or not $-2>3$ and $a\%2==1$

γ. $a>10$ or $b<3$ and $2*b>4$

δ. $a>12$ and $b<10$ or $3*-b>9$

ε. $a-\text{abs}(b)/5 * 2 > 18$

Μονάδες 10

ΘΕΜΑ 2

2.1. Οι τιμές που θα εμφανιστούν είναι:

4
12.0
5
Python
7
-3
3.2
6
0
True

2.2.

- α) ar, tet, riza
- β) %, **
- γ) int, abs
- δ) sqrt
- ε) ar<0, ar%2==0

2.3.

- α. False
- β. True
- γ. False
- δ. False
- ε. False

ΘΕΜΑ 4

Ένα κλιμάκιο του ΕΟΔΥ διενεργεί δωρεάν μαζικούς δειγματοληπτικούς ελέγχους για COVID-19 στη Κεντρική Πλατεία του Δήμου σας, κάθε Τρίτη από τις 11:00 έως 14:00. Προσέρχονται πολίτες όλων των ηλικιών προκειμένου να ελέγξουν εάν έχουν προσβληθεί από τον ιό Covid-19 κάνοντας RapidTest. Κατά την προσέλευση των πολιτών για έλεγχο, καταγράφονται από το κλιμάκιο του ΕΟΔΥ για στατιστικούς λόγους:

1. η ηλικία
2. το φύλο (‘Α’: άνδρας, ‘Γ’: γυναίκα)
3. το ΑΜΚΑ του πολίτη
4. και το αποτέλεσμα του RapidTest (με τιμές ‘Θ’: θετικό, ‘Α’:αρνητικό)

Να γραφεί πρόγραμμα σε Python το οποίο:

4.1 Να διαβάζει για κάθε πολίτη που προσέρχεται για έλεγχο:

α) την ώρα προσέλευσης του, ως ένα τετραψήφιο ακέραιο αριθμό (για παράδειγμα ο αριθμός 1115 αντιπροσωπεύει την ώρα 11:15), β) την ηλικία του, γ) το φύλο του δ) τον ΑΜΚΑ του και ε)το αποτέλεσμα του ελέγχου. Σε καμία τιμή εισόδου δεν απαιτείται έλεγχος εγκυρότητας.

Η καταχώρηση των στοιχείων σταματά όταν δοθεί ως ώρα προσέλευσης τιμή εκτός του διαστήματος 11:00- 14:00 ή όταν έχουν ελεγχθεί 250 πολίτες.

Μονάδες 7

4.2 Να εμφανίζει τη μικρότερη ηλικία γυναίκας που ελέγχθηκε τη συγκεκριμένη ημέρα. Θεωρήστε ότι προσέρχεται τουλάχιστον μια γυναίκα για έλεγχο τη συγκεκριμένη ημέρα.

Μονάδες 7

4.3 Να υπολογίζει και εμφανίζει για τη συγκεκριμένη ημέρα το συνολικό πλήθος των πολιτών που ελέγχθηκαν, πόσοι ήταν θετικοί, ποιο ήταν το ποσοστό των θετικών δειγμάτων και ποιος ο μέσος όρος ηλικίας των πολιτών που βρέθηκαν θετικοί, ως εξής: π.χ. Διενεργήθηκαν 250rapidtest και ανευρέθηκαν 3 θετικά (ποσοστό 1.2 %), με μέσο όρο ηλικίας τα 28.84 έτη.

Μονάδες 11

ΘΕΜΑ 4

```
1  #Αρχικοποίηση μεταβλητών
2  mikro=150      #μικρότερη ηλικία
3  pl_politon=pl_th=s_th=0 #4.3 πλήθος πολιτών, 4.3 πλήθος θετικών, 4.3 αθρ. ηλικιών θετικών
4  #Διάβασμα ώρας προσέλευσης
5  wra=int(input('Δώσε ώρα προσέλευσης ακέραια τιμή(11:00-14:00)'))
6  #έλεγχος αν είναι η ώρα προσέλευσης είναι από 11:00-14:00 και αν είναι κάτω από 250 πολίτες
7  while (wra>=1100 and wra<=1400) and (pl_politon<250):      #4.1
8      #4.1 εισαγωγή δεδομένων ηλικία, φύλο, ΑΜΚΑ, αποτέλεσμα test
9      ilikia=int(input('Δώσε την ηλικία '))
10     f=raw_input('Δώσε το φύλο (Α ή Γ')
11     amka=raw_input('Δώσε τον ΑΜΚΑ')
12     test=raw_input('Δώσε το αποτέλεσμα του Rapid test (Θ ή Α)')
13
14     pl_politon=pl_politon+1 #4.3 πλήθος ατόμων που έκαναν test
15
16     if test=='Θ':      #4.3, Αν το αποτέλεσμα ήταν Θετικό
17         pl_th=pl_th+1 #4.3 πλήθος θετικών
18         s_th=s_th+ilikia #4.3 άθροισμα ηλικίας θετικών
19
20         if f=='Γ':      #4.2, Έλεγχος αν είναι Γυναίκα.
21             if ilikia<mikro: #4.2 έλεγχος για μικρότερη ηλικία
22                 mikro=ilikia
23
24     wra=int(input('Δώσε επόμενη ώρα προσέλευσης ακέραια τιμή(11:00-14:00)'))
25
26     print 'Η μικρότερη ηλικία γυναίκας που ελέγχθηκε είναι ', mikro #4.2
27
28     if pl_th!=0:
29         milikia_th=float(s_th)/pl_th #4.3
30     else:
31         print 'Δεν υπήρξαν θετικά αποτελέσματα'
32         milikia_th=0
33
34     pososto_th=float(pl_th)/pl_politon*100 #4.3 ποσοστο θετικών
35     # 4.3 Εμφάνιση αποτελεσμάτων δειγματοληπτικών ελέγχων
36     print 'Διενεργήθηκαν ',pl_politon,' rapid test και ανευρέθηκαν ', pl_th
37     print ' θετικά (ποσοστό ', pososto_th,'%') με μέσο όρο ηλικίας ', milikia_th,' έτη.'
```

ΘΕΜΑ 2

2.1. Να γράψετε συνάρτηση με όνομα `count_A` η οποία θα δέχεται ένα όνομα (με ΚΕΦΑΛΑΙΑ) και θα επιστρέφει πόσες φορές εμφανίζεται το γράμμα Α μέσα στο όνομα π.χ. αν δεχθεί το όνομα ΑΜΑΛΙΑ θα επιστρέψει τον αριθμό 3.

Μονάδες 6

2.2. Δίνεται το παρακάτω τμήμα προγράμματος σε Python:

```
1  import math
2  x=int(input('Δώσε ένα αριθμό'))
3  if x <=10 or x >=99:
4      y=abs(x) * 2
5  else:
6      riza=math.sqrt(x)
7      print riza
8  print x%2!=0
```

Να γράψετε στο γραπτό σας:

- α. Τις μεταβλητές
- β. Τους λογικούς τελεστές
- γ. Τους αριθμητικούς τελεστές
- δ. Τις συνθήκες
- ε. Τους συγκριτικούς τελεστές
- στ. Τις ενσωματωμένες συναρτήσεις της Python που υπάρχουν στο παραπάνω τμήμα προγράμματος.

Μονάδες 7

2.3. Το παρακάτω τμήμα προγράμματος γράφτηκε ώστε να δημιουργεί και να εμφανίζει τη λίστα L με τα εξής περιεχόμενα:

`L=[2,4,8,16,32,64,128,256,512,1024]`

```
1     L=[]
2     x= (1)
3     for i in range ( (2) ):
4         (3) .append( (4) )
5         x= (5) * 2
6     print (6)
```

Στο τμήμα προγράμματος υπάρχουν έξι (6) κενά, τα οποία έχουν αριθμηθεί και υπογραμμιστεί. Να γράψετε στο γραπτό σας τον αριθμό του κενού και δίπλα τι πρέπει να συμπληρωθεί, ώστε το τμήμα προγράμματος να εκτελεί σωστά τη λειτουργία του.

Μονάδες 12

ΕΝΔΕΙΚΤΙΚΕΣ ΛΥΣΕΙΣ

ΘΕΜΑ 2

2.1

```
1  def count_A(X):
2      p=0
3      for i in X:
4          if i=='A':
5              p=p+1
6      return p
```

2.2

- α. x, y, riza
- β. or
- γ. *, %
- δ. $x \geq 10$, $x \leq 99$, $x \% 2 \neq 0$
- ε. $\geq$, $\leq$, $\neq$
- στ. int, abs

2.3

- 1. 2
- 2. 10
- 3. L
- 4. x
- 5. x
- 6. L

ΘΕΜΑ 4

Το διοικητικό συμβούλιο μιας εταιρείας αποφασίζει να κάνει αύξηση στους μισθούς των υπαλλήλων της σύμφωνα με τον παρακάτω πίνακα:

Μισθός σε €	Ποσοστό Αύξησης
450 - 750	15%
751 - 900	10%
901 – 1350	5%
>1350	2%

Για το λόγο αυτό να γράψετε πρόγραμμα σε γλώσσα προγραμματισμού Python που:

4.1 Να διαβάζει το ονοματεπώνυμο κάθε ενός από τους υπαλλήλους της εταιρείας και τον μισθό τους επαναληπτικά μέχρι να διαβάσει ως ονοματεπώνυμο το «TELOS». Θεωρείστε ότι δε θα δοθεί μισθός μικρότερος από 450 ευρώ.

Μονάδες 4

4.2 Να καλεί την συνάρτηση `extra_poso` (την οποία θα πρέπει να δημιουργήσετε), που θα δέχεται τον μισθό του υπαλλήλου και θα υπολογίζει και θα επιστρέφει στο κυρίως πρόγραμμα το ποσό της αύξησης που δικαιούται ο κάθε υπάλληλος, σύμφωνα με τον παραπάνω πίνακα.

Μονάδες 12

4.3. Το κυρίως πρόγραμμα να υπολογίζει και να εμφανίζει, μετά από κατάλληλο μήνυμα, τον νέο μισθό του υπαλλήλου και το ποσό της αύξησης που πήρε.

Μονάδες 4

4.4 Να υπολογίζει και να εμφανίζει το ποσοστό των υπαλλήλων που πήραν αύξηση πάνω από 40 €.

Μονάδες 5

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΕΝΔΕΙΚΤΙΚΕΣ ΛΥΣΕΙΣ

ΘΕΜΑ 4

```
1      def extra_poso (p):          #4.2 Συνάρτηση υπολογισμού αύξησης μισθού
2          if p>=450 and p<=750:
3              epipleon=p*(15/100.)    #epipleon το ποσό της αύξησης
4          elif p>=751 and p<=900:
5              epipleon=p*(10/100.)
6          elif p>=901 and p<=1350:
7              epipleon=p*(5/100.)
8          else:
9              epipleon=p*(2/100.)
10         return epipleon
11
12     #main program
13     pl=0; pl40=0          #4.4 pl=το πλήθος των ατόμων pl40=πλήθος με αύξηση πάνω από 40€
14     onep=raw_input('Δώσε ονοματεπώνυμο υπαλλήλου') #4.1 ονοματεπώνυμο υπαλλήλου
15     while onep!='TELOS':          #4.1
16         pl=pl+1                  #4.4 πλήθος υπαλλήλων
17         misthos=float(input('Δώσε τον μισθό του υπαλλήλου')) #4.1 διάβασμα μισθού
18         poso_aux=extra_poso(misthos)    #4.2 κλήση συνάρτ. extra_poso για υπολογισμό αύξησης
19         print 'Ο συνολικός μισθός του ',onep,' είναι ',misthos+poso_aux, '€' #4.3
20         print 'Η αύξηση του ', onep,' είναι: ',poso_aux, '€' #4.3
21         if poso_aux>40:          #4.4 Υπολογισμός ατόμων με αύξηση πάνω από 40€
22             pl40=pl40+1          #4.4
23             onep=raw_input('Δώσε επόμενο ονοματεπώνυμο υπαλλήλου ή TELOS για ΤΕΛΟΣ') #4.1
24     if pl!=0:                    #4.4 Έλεγχος αν διάβασε στοιχεία
25         pososto=pl40/float(pl)*100.    #4.4 υπολογισμός ποσοστού
26         print 'Το ποσοστό των υπαλλήλων που πήραν αύξηση πάνω από 40€ είναι: ',pososto, '%' #4.4
27     else:
```

28 `print 'Δεν διαβάστηκαν δεδομένα'`

ΘΕΜΑ 2

2.1. Δίνεται το παρακάτω τμήμα προγράμματος στη γλώσσα προγραμματισμού Python. Τι τιμές θα εμφανιστούν κατά την εκτέλεση του;

```
1   a='Καλή'
2   b='Άνοιξη'
3   msg=a+' '+b
4   for i in range(1,len(msg),3):
5       print i
6   print len(msg)
```

Μονάδες 10

2.2. Να τροποποιηθεί το παρακάτω τμήμα προγράμματος σε γλώσσα προγραμματισμού Python, ώστε να εμφανίζει διαδοχικά τους αριθμούς 96, 93, 90, 87, ..., 3 :

```
1   for i in range (3, 99, 3):
2       print i
```

Μονάδες 5

2.3. Να ξαναγράψετε το παρακάτω τμήμα προγράμματος, χρησιμοποιώντας την εντολή επανάληψης **while** αντί της εντολής επανάληψης **for** έτσι ώστε να δίνει το **ίδιο** αποτέλεσμα.

```
1   s=0
2   for i in range(1,10,2):
3       s=s+i
4       print s
5   print s
```

Μονάδες 10

ΘΕΜΑ 2

2.1. Θα εκτυπωθούν οι τιμές:

1
4
7
10
12

2.2. Η τροποποίηση του κώδικα:

```
1     for i in range (96, 0, -3):  
2         print i
```

2.3. Η μετατροπή σε while θα είναι όπως παρακάτω:

```
1     s=0  
2     i=1  
3     while i<10:  
4         s=s+i  
5         print s  
6         i=i+2  
7     print s
```

ΘΕΜΑ 4

Στο ακριτικό χωριό **Σύννορα** έχουν απογραφεί, με την τελευταία απογραφή, 380 άτομα. Να γραφεί πρόγραμμα σε γλώσσα Python, που θα χρησιμοποιηθεί για την νέα απογραφή έτσι ώστε να εξαχθούν κάποια απαραίτητα στατιστικά, το οποίο:

4.1. Να διαβάσει για **κάθε** κάτοικο την ηλικία του μέχρι να δοθεί αρνητικός αριθμός ή 0. Δεν χρειάζεται έλεγχος έγκυρης τιμής. Θεωρείστε ότι στην νέα απογραφή δίνεται τουλάχιστον μια ηλικία.

Μονάδες 4

4.2. Να υπολογίζει πόσα άτομα απογράφηκαν και αν υπάρχει μείωση ή αύξηση από την προηγούμενη απογραφή καθώς και πόσο ήταν αυτή, μετά από κατάλληλο μήνυμα.

Μονάδες 8

4.3. Να υπολογίζει και να εμφανίζει τον μέσο όρο ηλικίας των κατοίκων του χωριού, με την νέα απογραφή.

Μονάδες 5

4.4. Να υπολογίζει και να εμφανίζει την μικρότερη και την μεγαλύτερη ηλικία που έχει απογραφεί στην νέα απογραφή.

Μονάδες 8

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
    print b
```


ΘΕΜΑ 4

```
1  #4.2 Αρχικοποιούμε τον μετρητή των κατοίκων της νέας απογραφής
2  count=0
3  #4.2 Αρχικοποιούμε τον αθροιστή των ηλικιών
4  sum=0
5  #4.3 Αρχικοποιώ τις μεταβλητές min, max
6  #min, max η ελάχιστη και μέγιστη ηλικία της νέας απογραφής
7  min=1000
8  max=0
9  #4.1 Ο χρήστης δίνει τις ηλικίες των κατοίκων
10 age=int(input('Δώσε την ηλικία κατοίκου (>0): '))
11 #4.1 Επανάληψη μέχρι να δοθεί αρνητικός αριθμός
12 while age>0:
13     count=count+1 #4.2 για τα άτομα της νέας απογραφής
14     sum=sum+age
15 #4.3 Υπολογίζω τα min, max
16     If (min>age):
17         min=age
18     If (max<age):
19         max=age
20     age=int(input('Δώσε την ηλικία άλλου κατοίκου (>0): '))
21
22 #4.2 Εμφάνιση κατοίκων νέας απογραφής
23 print 'Ο αριθμός κατοίκων με την νέα απογραφή είναι:', count
24 #4.2 Υπολογισμός και εμφάνισης αύξησης ή μείωσης
25 if count>380:
26     print 'έχουμε αύξηση: ', count-380, 'κατοίκων'
27 elif count<380:
28     print 'έχουμε μείωση: ', 380-count, 'κατοίκων'
29 else:
30     print 'Ίδιος αριθμός κατοίκων'
31
32 #4.2 Βρίσκουμε και εκτυπώνουμε τον μέσο όρο των ηλικιών
33 mesos_oros=float(sum)/count
34
35 print 'Ο μέσος όρος ηλικιών είναι: ',mesos_oros
36
37 #4.3 Εκτυπώνουμε τα αποτελέσματα
38 print 'Η μικρότερη ηλικία είναι : ', min
39 print 'Η μεγαλύτερη ηλικία είναι: ', max
```

ΘΕΜΑ 2

2.1. Να συμπληρώσετε την παρακάτω παράγραφο, που περιγράφει τον Αντικειμενοστρεφή Προγραμματισμό (Object-oriented programming - OOP), με τα σωστά λεκτικά που δίνονται ακολούθως και στη σωστή πτώση.

Πρόκειται για είδος προγραμματισμού που περιστρέφεται γύρω από την έννοια ...(Α).... η οποία περιγράφει(Β)...., τα οποία περιέχουν δεδομένα στη μορφή(Γ)....και κώδικα στη μορφή...(Δ)....

Αντικείμενο (Object),

Κλάση (Class),

Μέθοδοι (Methods)

Ιδιότητες (Properties),

Μονάδες 12

2.2. Να γράψετε στο γραπτό σας τι θα εμφανίσει η εκτέλεση των παρακάτω τμημάτων προγραμμάτων. Δικαιολογείστε τις απαντήσεις σας.

Τμήμα προγράμματος 1	Τμήμα προγράμματος 2
<pre>1 A =0 2 while A>5: 3 print A 4 A =A – 2 5 print “Τέλος επανάληψης”</pre>	<pre>1 for i in range (0,100,100): 2 print i</pre>
(Μονάδες 6)	(Μονάδες 7)

Μονάδες 13

ΘΕΜΑ 2

2.1.

A - της Κλάσης (Class)

B - Αντικείμενα (Objects)

Γ - Ιδιοτήτων (Properties)

Δ - Μεθόδων (Methods)

2.2.

1. **Τέλος επανάληψης.** Δεν ικανοποιείται η συνθήκη στην επανάληψη και δεν εκτελεί ποτέ τις εντολές του βρόχου.
2. **0.** Ο βρόχος εκτελείται μόνο μια φορά, όταν το i είναι 0. Η επόμενη τιμή είναι $0+100 = 100$ εκτός του range.

ΘΕΜΑ 4

Μια ομάδα μπάσκετ έδωσε στην Euroleague 12 αγώνες.

Να αναπτύξετε πρόγραμμα στην γλώσσα προγραμματισμού Python το οποίο:

4.1. Να διαβάζει τους πόντους που πέτυχε η ομάδα σε κάθε αγώνα. Οι πόντοι αυτοί να είναι θετικός αριθμός.

Μονάδες 7

4.2. Για κάθε αγώνα, να εμφανίζει το μήνυμα «Υπέροχα» αν η επίδοση της ομάδας είναι άνω των 85 πόντων.

Μονάδες 8

4.3. Να εμφανίζει το μέσο όρο των πόντων που πέτυχε η ομάδα στους 12 αυτούς αγώνες.

Μονάδες 10

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1 s=0 #4.3, αρχικοποίηση μετρητή
2 #εντολή επανάληψης για το διάβασμα πόντων κάθε αγώνα
3 #και έλεγχο επίδοσης
4 for i in range (12) #4.1, 4.2
5     x=int(input('Δώσε πόντους')) #4.1, εισαγωγή πόντων
6     while x<=0: #4.1, έλεγχος αν οι πόντοι είναι θετικός αριθμός
7         print "έδωσες αρνητικό αριθμό" #4.1, μήνυμα λάθους
8         x=int(input('Δώσε ξανά σωστούς πόντους:')) #4.1, επανεισαγωγή
9     s=s+x #4.3, άθροιση πόντων για μέσο όρο
10    if x>85: #4.2, έλεγχος επίδοσης
11        print "Υπέροχα" #4.2, μήνυμα υψηλής επίδοσης
12    mo=s/12. #4.3, εύρεση μέσου όρου
13    print mo #4.3, εκτύπωση μέσου όρου
```

ΘΕΜΑ 2

2.1. Δίνονται οι τιμές των μεταβλητών **A=5, B=3, C=4, D=2** και η παρακάτω σύνθετη λογική έκφραση:

$\text{not}(\text{A}+\text{B}*3>15) \text{ or } (\text{C}*4 \% 2 == \text{B}^{}(\text{C}-2) \text{ and } (\text{C} == 8 / \text{D}))$**

Να αντικαταστήσετε τις μεταβλητές με τις τιμές που δίνονται, να εκτελέσετε τις αριθμητικές πράξεις, να κάνετε τις απαραίτητες συγκρίσεις και να εκτελέσετε τις λογικές πράξεις, ώστε να υπολογίσετε την τελική τιμή της έκφρασης.

Μονάδες 5

2.2. Να γράψετε στο τετράδιό σας τους αριθμούς 1,2,3,4,5 της **Στήλης Α** και δίπλα το γράμμα α, β, γ, δ, ε της **Στήλης Β**, που δίνει τη σωστή αντιστοιχία.

Στήλη Α	Στήλη Β
1. $X==2$:	α. int
2. $\text{if } X\%2!=0$	β. Έλεγχος για μη πολλαπλάσιος του 2
3. $\text{if } X\%2==0$	γ. Συνθήκη
4. $\text{done}=\text{True}$	δ. Έλεγχος για άρτιος
5. $X=4$	ε. boolean

Μονάδες 10

2.3. Δίνεται ο παρακάτω κώδικας σε γλώσσα προγραμματισμού Python. Τι τιμές θα εμφανιστούν κατά την εκτέλεση του αν δοθούν οι τιμές:

α) 80, β) -100, γ) 0, δ) 150

```
1  v=int(input('Δώσε βαθμολογία'))
2  if v <=100:
3      if v <=85:
4          if v <=70:
5              if v <=55:
6                  print 'ΑΠΟΤΥΧΙΑ'
7              else:
8                  print 'C'
9          else:
10             print 'B'
11     else:
12         print 'A'
13 else:
14     print 'Έδωσες λάθος βαθμό'
```

Να γράψετε στο τετράδιο σας το α, β, γ ή δ και δίπλα την τιμή που θα εμφανιστεί σύμφωνα με τις παραπάνω τιμές εισόδου.

Μονάδες 10

ΘΕΜΑ 2

2.1. True

$A+B*3=14$ άρα η $A+B*3 > 15$ είναι False και με το not μπροστά γίνεται True, οπότε η παράσταση ολόκληρη είναι επίσης **True** αφού έχουμε τον τελεστή **or** να συνδέει τις δύο προτάσεις.

2.2.

1 $\rightarrow \gamma$

2 $\rightarrow \beta$

3 $\rightarrow \delta$

4 $\rightarrow \epsilon$

5 $\rightarrow \alpha$

2.3.

Είσοδος	Έξοδος
80	B
-100	ΑΠΟΤΥΧΙΑ
0	ΑΠΟΤΥΧΙΑ
150	Έδωσες λάθος βαθμό

ΘΕΜΑ 4

Σε ένα εργοστάσιο κατασκευής τρισδιάστατων εκτυπωτών απασχολούνται 35 υπάλληλοι. Ο μηνιαίος μισθός του **κάθε** υπαλλήλου προκύπτει με τον παρακάτω τρόπο:

- ο κάθε υπάλληλος έχει βασικό μισθό 720 €,
- για κάθε ώρα υπερωρίας που εργάστηκε μέσα στον μήνα αμείβεται με 5 € ανά ώρα,
- για κάθε τρία χρόνια εργασίας λαμβάνει χρονοεπίδομα 25 € ανά τριετία, π.χ. εάν κάποιος υπάλληλος έχει 10 χρόνια εργασίας, θα λάβει $3 * 25 = 75$ € επίδομα,
- ο άθροισμα όλων των παραπάνω δημιουργούν τον **ακαθάριστο μισθό του υπαλλήλου**.

Στον **ακαθάριστο μισθό** του κάθε υπαλλήλου, που υπολογίζεται όπως παραπάνω, υπάρχουν οι παρακάτω κρατήσεις:

- 8% για το ταμείο της ασφάλισης του υπαλλήλου, και
- το 7% για την σύνταξη του υπαλλήλου.

Να αναπτύξετε πρόγραμμα στην γλώσσα προγραμματισμού Python το οποίο

4.1. Να διαβάζει για **κάθε** υπάλληλο το ονοματεπώνυμο, τον αριθμό υπερωριών που εργάστηκε κατά τον μήνα Μάρτιο καθώς και τα χρόνια εργασίας του υπαλλήλου. Τα χρόνια εργασίας να είναι μη αρνητικός αριθμός.

Μονάδες 6

4.2. Να υπολογίζει τον ακαθάριστο μισθό του υπαλλήλου (όπως περιγράφεται παραπάνω) και θα τον εμφανίζει μετά το μήνυμα «Ο μικτός σου μισθός είναι».

Μονάδες 7

4.3. Να υπολογίζει και να εμφανίζει το ονοματεπώνυμο του υπαλλήλου και τις **καθαρές** αποδοχές του μετά το μήνυμα «Οι καθαρές σου αποδοχές είναι».

Μονάδες 8

4.4. Να υπολογίζει και να εμφανίζει το σύνολο των κρατήσεων από όλους τους υπαλλήλους μετά το μήνυμα «Οι συνολικές κρατήσεις των 35 υπαλλήλων είναι».

Μονάδες 4

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1. s_kratiseis=0 #4.4 αρχικοποιει τον αθροιστή για τις συνολικές κρατήσεις
2. for i in range (35):
3.     basikos=720.00 # εκχωρείται τιμή στον βασικό μισθό
4.     # 4.1, διαβάζει το ονοματεπώνυμο του υπαλλήλου
5.     onoma=raw_input('Δώσε ονοματεπώνυμο υπαλλήλου: ')
6.     #4.1 διαβάζει τον αριθμό των υπερωριών σε ώρες
7.     wres=int(input('Δώσε αριθμό υπερωριών σε ώρες: '))
8.     # 4.1, διαβάζει τα χρόνια εργασίας του υπαλλήλου
9.     xronia = int(input('Δώσε τα χρόνια εργασίας: '))
10.    #4.1 έλεγχος αν τα χρόνια εργασίας είναι μη αρνητικός αριθμός
11.    while xronia<0:
12.        print "Έδωσες αρνητικό αριθμό"
13.        xronia = int(input('Δώσε σωστά τα χρόνια εργασίας: '))
14.    #4.2 υπολογισμός υπερωριών
15.    m_wres=wres*5.00
16.    #4.2 υπολογισμός χρονοεπιδόματος
17.    m_xronos=25.00*(xronia/3)
18.    #4.2 υπολογισμός ακαθάριστου μισθού και εμφάνισή του
19.    salary=basikos+m_wres+m_xronos
20.    print 'Ο μικτός σου μισθός είναι: ', salary
21.    #4.3 υπολογισμός κρατήσεων
22.    t_asf=salary*8/100
23.    t_sin=salary*7/100
24.    kratiseis=t_asf+t_sin #αθροίζει τις κρατήσεις του υπαλλήλου
25.    s_kratiseis=s_kratiseis+kratiseis #4.4 Ενημερώνει τον αθροιστή
26.    #4.3 υπολογισμός καθαρών αποδοχών και εμφάνισή τους
27.    clean_salary=salary-kratiseis
28.    print 'Ο/Η ', onoma, 'οι καθαρές σου αποδοχές είναι: ', clean_salary
29. #4.4 εμφανίζει τις συνολικές κρατήσεις
30. print 'Οι συνολικές κρατήσεις των 35 υπαλλήλων είναι: ', s_kratiseis
```

ΘΕΜΑ 2

2.1. Δίνεται το παρακάτω τμήμα προγράμματος σε γλώσσα προγραμματισμού Python:

```
1    x=24
2    while x!=0:
3        y=x%10
4        if y%2==0:
5            x=x+1
6            print x , y
7        else:
8            x=x/2
9            print y , x
```

Επίσης δίνεται το παρακάτω υπόδειγμα- πίνακας (πίνακας τιμών) :

	x	y	Έξοδος
Αρχικές Τιμές	24	?	
1 ^η Επανάληψη	...	...	...
....	...	...	...

Να μεταφέρετε στο γραπτό σας τον παραπάνω πίνακα και να τον συμπληρώσετε, προσθέτοντας μία νέα γραμμή στον πίνακα για κάθε επανάληψη του προγράμματος.

Μονάδες 12

2.2. Να γράψετε μια συνάρτηση με όνομα `check_num` η οποία θα δέχεται έναν ακέραιο αριθμό `x`, και θα επιστρέφει την τιμή `True` αν ο αριθμός είναι μονός ή την τιμή `False` αν ο αριθμός είναι ζυγός.

Μονάδες 6

2.3. Δίνεται το παρακάτω τμήμα προγράμματος σε γλώσσα προγραμματισμού Python, το οποίο υπολογίζει και τυπώνει το άθροισμα των θετικών ακεραίων αριθμών που είναι πολλαπλάσιοι του 3 και βρίσκονται στο διάστημα από το 100 έως το 999.

```
1    i = (1)
2    s = 0
3    while i (2) 999 :
4        if i (3) 3 == (4) :
5            s = (5) + i
6            i = i + (6) .
7    print " Το άθροισμα είναι: " , (7)
```

Στο τμήμα προγράμματος υπάρχουν υπογραμμισμένα κενά τα οποία έχουν αριθμηθεί. Να γράψετε στο τετράδιό σας τους αριθμούς 1 , 2 , 3 , 4 , 5 , 6 και 7 που αντιστοιχούν στα κενά του παραπάνω τμήματος προγράμματος και δίπλα σε κάθε αριθμό αυτό που πρέπει να συμπληρωθεί για να υλοποιείται σωστά το τμήμα προγράμματος.

Μονάδες 7

ΘΕΜΑ 2

2.1

	x	y	Έξοδος
Αρχικές Τιμές	24	?	
1 ^η Επανάληψη	25	4	25 4
2 ^η Επανάληψη	12	5	5 12
3 ^η Επανάληψη	13	2	13 2
4 ^η Επανάληψη	6	3	3 6
5 ^η Επανάληψη	7	6	7 6
6 ^η Επανάληψη	3	7	7 3
7 ^η Επανάληψη	1	3	3 1
8 ^η Επανάληψη	0	1	1 0

2.2

```
1  def check_num(x):  
2      if x%2==1:  
3          return True  
4      else:  
5          return False
```

2.3

1. 100
2. <=
3. %
4. 0
5. S
6. 1
7. S

ΘΕΜΑ 4

Οι μαθητές της Β' Λυκείου προκειμένου να μαζέψουν χρήματα αποφάσισαν να κάνουν ένα bazaar πουλώντας κάποια αναμνηστικά με σκοπό να συγκεντρώσουν ένα ποσό για την οικονομική ενίσχυση του σχολικού ταμείου για να πάνε μια τριήμερη εκδρομή στη Λίμνη Πλαστήρα. Να γράψετε πρόγραμμα σε Python το οποίο:

4.1. Να διαβάζει συνεχώς το ποσό των χρημάτων που δίνουν κάποιοι πολίτες για την αγορά αναμνηστικών μέχρι το συνολικό ποσό των αγορών που πρόκειται να αποταμιευθεί να ξεπεράσει τα 1.500 € .

Μονάδες 9

4.2. Εμφανίζει το συνολικό ποσό που συγκεντρώθηκε και το πλήθος των ατόμων που συνέβαλαν για την εκταμίευση του ποσού αυτού.

Μονάδες 6

4.3. Υπολογίζει και εμφανίζει το ποσοστό των πολιτών που έδωσαν παραπάνω από 50€.

Μονάδες 7

4.4. Υπολογίζει και εμφανίζει το μικρότερο ποσό που δόθηκε στο bazaar αγορών.

Μονάδες 3

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4.

```
1  synolo=0.0          #4.1 Αρχικοποίηση μεταβλητής synolo
2  atoma=0; pl50=0     #4.2 , 4.3 Αρχικοποίηση atoma που συμμετείχαν, pl50 άτομα με ποσό>50€
3  mikrotero=150000   #4.4 Αρχικοποίηση μεταβλητής mikrotero ποσό
4  while synolo<=1500:                                #4.1 Όσο synolo <1500 € δούλεψε
5      poso=input('Δώσε το ποσό της αγοράς')          #4.1 Διάβασε ποσό αγοράς
6      synolo=synolo+poso                               #4.1 Υπολόγισε το μέχρι στιγμής synolo
7      atoma=atoma+1                                    #4.2 τα άτομα που συμμετείχαν
8      if poso<mikrotero:                               #4.4 Εύρεση ελαχίστου
9          mikrotero=poso                               #4.4
10     if poso>50:                                       #4.3 Εύρεση πόσοι έδωσαν πάνω από 50€
11         pl50=pl50+1                                  #4.3
12     print 'Το συνολικό που συγκεντρώθηκε είναι',synolo,'€' #4.2
13     print 'και συνέβαλαν',atoma,'άτομα για να συγκεντρωθεί αυτό το ποσό' #4.2
14     pososto=(pl50/float(atoma))*100.                #4.3
15     print 'Το ποσοστό των πολιτών που έδωσαν πάνω από 50 € είναι',pososto,'% ' #4.3
16     print 'Το μικρότερο ποσό που δόθηκε στο bazaar είναι',mikrotero,'€' #4.4
```


ΘΕΜΑ 2

2.1. Να συμπληρώσετε τον παρακάτω πίνακα με τις σωστές λογικές τιμές.

X	Y	not X	X and Y
True	True	False	2
True	False	1	False
False	True	True	3
False	False	True	4

Να γράψετε στο γραπτό σας τον αριθμό (1 έως 4) που αντιστοιχεί στο κενό κελί του πίνακα και δίπλα τη σωστή ένδειξη **True** ή **False**.

Μονάδες 4

2.2. Να γράψετε στο γραπτό σας τους αριθμούς που θα εμφανίσει η εκτέλεση των παρακάτω τμημάτων προγραμμάτων.

Τμήμα προγράμματος 1	Τμήμα προγράμματος 2
<pre>1 A=10 2 while A>5: 3 print A 4 A = A - 1</pre>	<pre>1 for i in range (0,100,20): 2 print i print i</pre>
(Μονάδες 6)	(Μονάδες 7)

Μονάδες 13

2.3. Να μετατρέψετε, έτσι ώστε να δίνει τα ίδια αποτελέσματα, το παραπάνω **Τμήμα προγράμματος 1**, χρησιμοποιώντας την επαναληπτική δομή **for**.

Μονάδες 8

QEMA 2

2.1.

1. False
2. True
3. False
4. False

2.2.

A) 10,9,8,7,6

B) 0,20,40,60,80

2.3.

```
1 for A in range (10,5,-1):  
2     print A
```

ΘΕΜΑ 4

Μια αεροπορική εταιρεία πρόκειται να προσφέρει ειδικό επίδομα (bonus) στους 300 υπτάμενους φροντιστές (αεροσυνοδούς) που απασχολεί. Το bonus υπολογίζεται σαν ένα ποσοστό των ετήσιων αποδοχών τους, ανάλογα με τα μίλια πτήσεων που πραγματοποίησαν συνολικά κατά το περασμένο έτος, με βάση τον παρακάτω πίνακα:

Ετήσια μίλια πτήσεων	Ποσοστό bonus (επί των ετήσιων αποδοχών)
200000-250000	5%
250001 - 300000	10%
>300000	15%

Οι υπτάμενοι φροντιστές που πραγματοποίησαν λιγότερα από 200000 μίλια πτήσεων ετησίως, δε δικαιούνται bonus. Ο υπολογισμός του bonus δεν γίνεται κλιμακωτά.

Να γραφεί πρόγραμμα σε Python που:

4.1 Για κάθε υπτάμενο φροντιστή διαβάζει τον κωδικό εργαζομένου του (EMPL), τα μίλια πτήσεων (MIL) που συμπλήρωσε κατά το περασμένο έτος και τις ετήσιες αποδοχές του σε ευρώ (SAL), χωρίς να πραγματοποιείται έλεγχος ορθότητας των πληκτρολογούμενων δεδομένων.

Μονάδες 4

4.2 Υπολογίζει το ποσό του bonus που δικαιούται ο συγκεκριμένος υπτάμενος φροντιστής, με βάση τον παραπάνω πίνακα και το εμφανίζει, εφόσον δεν είναι μηδενικό. Αν ο υπτάμενος φροντιστής δε δικαιούται bonus, εμφανίζει το μήνυμα “Ο υπτάμενος φροντιστής δε δικαιούται bonus!”

Μονάδες 12

4.3 Βρίσκει κι εμφανίζει τον κωδικό εργαζομένου του φροντιστή αέρος με τα περισσότερα ετήσια μίλια πτήσης, καθώς και πόσα ήταν τα ετήσια μίλια του. Θεωρούμε πως είναι μοναδικός.

Μονάδες 5

4.4 Υπολογίζει κι εμφανίζει το συνολικό ποσό που θα διαθέσει η αεροπορική εταιρεία για τα bonus των υπτάμενων φροντιστών.

Μονάδες 4

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΕΝΔΕΙΚΤΙΚΕΣ ΛΥΣΕΙΣ

ΘΕΜΑ 4

```
1  MAX_MIL = -1 # 4.3 Αρχικοποίηση της τιμής των μέγιστων ετήσιων μιλίων
2  sum = 0.0 # 4.4 Αρχικοποίηση του συνολ. ποσού για τα bonus
3  # Ερώτημα 4.1: Εισαγωγή δεδομένων για τους 300 ιπτ. φροντιστές
4  for x in range(300):
5      EMPL = raw_input("Δώσε τον κωδικό του ιπτάμενου φροντιστή:")
6      MIL = int(input("Δώσε τα ετήσια μίλια του:"))
7      SAL = float(input("Δώσε τις ετήσιες αποδοχές του σε ευρώ:"))
8      # Ερώτημα 4.2: Υπολογισμός του bonus που δικαιούται ο φροντιστής
9      if MIL < 200000:
10         bonus = 0.0
11     elif MIL <= 250000:
12         bonus = 5 / 100. * SAL
13     elif MIL <= 300000:
14         bonus = 10 / 100. * SAL
15     else:
16         bonus = 15 / 100. * SAL
17     if bonus != 0:
18         print "Ο ιπτάμενος φροντιστής θα πάρει bonus:", bonus
19     else:
20         print "Ο ιπτάμενος φροντιστής δε δικαιούται bonus!"
21     # 4.3: Αν τα ετήσια μίλια του ιπτ. φροντιστή είναι περισσότερα
22     # από το τρέχον μέγιστο, ενημερώνονται και τα μέγιστα μίλια
23     # αλλά και ο κωδικός του φροντιστή που τα πραγματοποίησε
24     if MIL > MAX_MIL:
25         MAX_MIL = MIL
26         MAX_EMPL = EMPL
27     sum = sum + bonus # 4.4 Πρόσθεση του bonus του φροντιστή στο σύνολο
28 # Ερώτημα 4.3:
29 print "Κωδικός ιπτ. φροντιστή με τα περισσότερα ετήσια μίλια:", MAX_EMPL
30 print " και τα ετήσια μίλια του ήταν ", MAX_MIL
31 # Ερώτημα 4.4:
32 print "Το συνολικό ποσό που θα διαθέσει η εταιρεία στα bonus είναι: ", sum
```

ΘΕΜΑ 2

2.1.

A. Να γράψετε ισοδύναμο με το παρακάτω τμήμα προγράμματος Python, χρησιμοποιώντας την εντολή επανάληψης `while` αντί της εντολής επανάληψης `for`.

```
1  gin= 1
2  for x in range(10) :
3      gin= gin * x
4      print x
5  print gin
```

Μονάδες 5

B. Πόσες φορές θα εκτελεστεί η εντολή 3 (`gin=gin*x`);

Μονάδα 1

Γ. Ποιες τιμές θα εμφανιστούν κατά τη διάρκεια εκτέλεσης του παραπάνω τμήματος προγράμματος;

Μονάδες 5

2.2.

Να αντιστοιχίσετε κάθε στοιχείο της στήλης A του ακόλουθου πίνακα, με το κατάλληλο στοιχείο της στήλης B. Δύο από τα στοιχεία της στήλης B περισσεύουν:

Στήλη A	Στήλη B
1. ==	A. Αριθμητικός τελεστής
2. pow	B. Συγκριτικός τελεστής
3. %	Γ. Λογικός Τελεστής
4. power	Δ. Ενσωματωμένη συνάρτηση
	E. Μη ενσωματωμένη συνάρτηση
	ΣΤ. Όνομα μεταβλητής

Μονάδες 4

2.3

Στο παρακάτω τμήμα προγράμματος σε Python υπάρχουν πέντε (5) κενά, τα οποία έχουν αριθμηθεί και υπογραμμιστεί. Να γράψετε στο γραπτό σας τον αριθμό του κάθε κενού και δίπλα τι πρέπει να συμπληρωθεί, ώστε το τμήμα προγράμματος να υπολογίζει και να εμφανίζει το άθροισμα των περιττών ακεραίων από το 100 έως το 200.

```
1  sum= ...(1)...
2  for x in range( ...(2)... , ...(3)... ) :
3      if x ...(4)... :
4          sum= sum + ...(5)...
5  print "Το άθροισμα είναι:", sum
```

Μονάδες 10

ΕΝΔΕΙΚΤΙΚΕΣ ΛΥΣΕΙΣ

ΘΕΜΑ 2

2.1

A.

```
1  gin= 1
2  x=0
3  while x <10 :
4      gin= gin * x
5      print x
6      x=x+1
7  print gin
```

B. Η εντολή ***gin=gin*x*** θα εκτελεστεί 10 φορές

Γ. Κατά την διάρκεια της εκτέλεσης του προγράμματος θα εμφανιστούν οι τιμές:

0, 1, 2, 3, 4, 5, 6, 7 , 8, 9,0

2.2

1. B

2. Δ

3. A

4. ΣΤ

2.3

(1): 0

(2): 100

(3): 200

(4): % 2 == 1

(5): x

ΘΕΜΑ 2

2.1.

Δίνονται οι τιμές των μεταβλητών $a = -5$, $b = 5$, $c = 3$ και $d = 9$. Να βρείτε, δείχνοντας αναλυτικά τα βήματα που ακολουθείτε για τη λύση, αν η έκφραση που ακολουθεί είναι Αληθής (True), ή Ψευδής (False):

`not((b % 5 == 0) and ( c**2 +a < d/2 ))`

Μονάδες 10

2.2

Το ακόλουθο πρόγραμμα σε Python έχει γραφεί για να διαβάσει 5 θετικούς αριθμούς και να υπολογίζει και να εμφανίζει το γινόμενο όσων από αυτούς είναι μεγαλύτεροι του 10 και πολλαπλάσιοι του 3 (συγχρόνως). Στο πρόγραμμα, όμως, υπάρχουν πέντε (5) λάθη. Για κάθε ένα από τα λάθη να γράψετε στο γραπτό σας τον αριθμό της γραμμής στην οποία το εντοπίσατε, να περιγράψετε το λάθος και να δηλώσετε το είδος του λάθους:

```
1  p=0
2  for i in range(1,5):
3      x=int(input("Δώσε ένα θετικό ακέραιο:"))
4      if x > 10 or x mod 3 == 0:
5          p= p*x
6  print "ΓΙΝΟΜΕΝΟ=", p
```

Μονάδες 15

ΕΝΔΕΙΚΤΙΚΕΣ ΛΥΣΕΙΣ

ΘΕΜΑ 2

2.1

Για $a = -5$, $b = 5$, $c = 3$ και $d = 9$:

$\text{not}((b \% 5 == 0) \text{ and } (c**2 + a < d/2))$

$\text{not}((5 \% 5 == 0) \text{ and } (3**2 + (-5) < 9/2))$

$\text{not}((5 \% 5 == 0) \text{ and } (9 + (-5) < 9/2))$

$\text{not}((0 == 0) \text{ and } (9 + (-5) < 4))$

$\text{not}((0 == 0) \text{ and } (4 < 4))$

$\text{not}(\text{Αληθής and Ψευδής})$

$\text{not}(\text{Ψευδής})$

Αληθής

Η έκφραση είναι Αληθής.

2.2

Αριθμός γραμμής	Περιγραφή λάθους	Είδος λάθους
1	Το γινόμενο πρέπει να αρχικοποιείται στην τιμή 1 κι όχι στην τιμή 0. Σωστό: $p=1$	Λογικό
2	Διαβάζονται 4 ακέραιοι κι όχι 5, όπως πρέπει. Σωστό: $\text{range}(1,6)$ ή $\text{range}(5)$	Λογικό
4	Για να ισχύουν ταυτόχρονα οι δύο περιορισμοί απαιτείται ο λογικός τελεστής and (κι όχι ο τελεστής or).	Λογικό
4	Ο τελεστής που δίνει το υπόλοιπο της ακέραιας διαίρεσης γράφεται % (κι όχι mod)	Συντακτικό
5	Η εντολή p=p*x περιλαμβάνεται στην if κι επομένως απαιτείται εσοχή	Συντακτικό

ΘΕΜΑ 4

Η μέγιστη επιτρεπόμενη τιμή της περιεκτικότητας κατ' όγκο των καυσαερίων σε οξείδια του αζώτου (NOx) ανηγμένη σε οξυγόνο αναφοράς 3%, σε ppm είναι 150. Σε περίπτωση που ο ρύπος αυτός ξεπεράσει τα 150 ppm πρέπει να ληφθούν μέτρα. Τα υπουργείο περιβάλλοντος έχει εγκαταστήσει 50 σταθμούς μέτρησης του ρύπου σε διάφορα σημεία της χώρας.

Να αναπτύξετε πρόγραμμα στην γλώσσα προγραμματισμού Python το οποίο

4.1. Να διαβάζει την αριθμητική τιμή του ρύπου **για κάθε** σταθμό μέτρησης. Η αριθμητική τιμή αυτή να είναι θετικός αριθμός.

Μονάδες 7

4.2. Να υπολογίζει και να εμφανίζει το **πλήθος** των σταθμών με ένδειξη **μεγαλύτερη** από 150.

Μονάδες 8

4.3. Να υπολογίζει και να εμφανίζει τη **μέση τιμή** του ρύπου στους 50 σταθμούς.

Μονάδες 10

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1 k=0 #4.2, αρχικοποίηση μετρητή
2 s=0 # 4.3, αρχικοποίηση αθροιστή
3 #εντολή επανάληψης για την μέτρηση ρύπων στους 50 σταθμούς
4 for i in range (0,50): #4.1
5     x=int(input('Δώσε τιμή ρύπου:')) #4.1, εισαγωγή ρύπου
6     #έλεγχος αρνητικού ρύπου και επαναεισαγωγή έως ότου δοθεί θετικός αριθμός
7     while x<=0: #4.1, έλεγχος
8         print "Έδωσες αρνητικό αριθμό" #4.1, μήνυμα προειδοποίησης
9         x=int(input('Δώσε ξανά σωστούς ρύπους:')) #4.1, επαναεισαγωγή ρύπου
10    s=s+x #4.3, άθροιση ρύπων για μέσο όρο
11    #έλεγχος και εύρεση αριθμού σταθμών για επικίνδυνες τιμές ρύπου
12    if x>150: #4.2, έλεγχος
13        k=k+1 #4.2, αύξηση του μετρητή
14    mo=s/50. #4.3, εύρεση μέσου όρου
15    print k #4.2, εκτύπωση αριθμού σταθμών με υψηλές τιμές ρύπων
16    print mo #4.3, εκτύπωση μέσου όρου
```

ΘΕΜΑ 4

Ο εκατομμυριοστός πελάτης ενός πολυκαταστήματος λαμβάνει από τη Διεύθυνση του πολυκαταστήματος ως δώρο το ποσό των 1000 ευρώ, το οποίο μπορεί να διαθέσει για αγορές εντός του πολυκαταστήματος.

Να γράψετε πρόγραμμα σε Python το οποίο:

4.1 Για κάθε είδος που επιθυμεί να αγοράσει ο πελάτης θα διαβάζει την ονομασία είδους και το κόστος αγοράς του σε ευρώ, χωρίς να απαιτείται έλεγχος ορθότητας των πληκτρολογούμενων δεδομένων. Αν το υπόλοιπο χρημάτων επαρκεί για την αγορά, τότε θα εμφανίζεται το μήνυμα "Επιτυχής αγορά" και θα αφαιρείται το κόστος αγοράς από το υπόλοιπο. Διαφορετικά θα εμφανίζεται το μήνυμα "Δεν επαρκεί το υπόλοιπο".

Μονάδες 8

4.2 Η διαδικασία των αγορών θα τερματίζεται είτε όταν ως ονομασία είδους δοθεί η λέξη "TELOS", είτε όταν μηδενισθεί το υπόλοιπο των χρημάτων.

Μονάδες 5

4.3 Το πρόγραμμα θα εμφανίζει την ονομασία του ακριβότερου είδους που αγοράστηκε, καθώς και το κόστος αγοράς του. Θεωρείστε ότι θα αγοραστεί τουλάχιστον ένα είδος.

Μονάδες 8

4.4 Στην περίπτωση που δεν ξοδεύτηκε όλο το ποσό, θα εμφανίζεται το ποσό που περίσσεψε. Διαφορετικά θα εμφανίζεται το μήνυμα "Μηδενικό υπόλοιπο".

Μονάδες 4

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΕΝΔΕΙΚΤΙΚΕΣ ΛΥΣΕΙΣ

ΘΕΜΑ 4

```
1  MAX_COST = -1 # 4.3 Αρχικοποίηση του μέγιστου κόστους αγοράς
2  YPOL = 1000 # Αρχικοποίηση του υπολοίπου
3
4  # Εισαγωγή ονομασίας 1ου είδους προς αγορά
5  EIDOS = raw_input('Δώσε την ονομασία του 1ου είδους:')
6
7  # Ερώτημα 4.2: Οι αγορές συνεχίζονται όσο υπάρχει υπόλοιπο και
8  # δεν έχει δοθεί ως ονομασία είδους η λέξη "TELOS"
9  while YPOL>0 and EIDOS!='TELOS':
10     COST = float(input('Δώσε το κόστος αγοράς σε ευρώ:'))
11     # 4.1: εξετάζουμε αν είναι εφικτή η αγορά
12     if COST <= YPOL:
13         print 'Επιτυχής αγορά'
14         YPOL = YPOL - COST
15         # Ερώτημα 4.3: Αν το κόστος αυτού του είδους είναι μεγαλύτερο
16         # από το τρέχον μέγιστο, ενημερώνεται και το μέγιστο κόστος,
17         # αλλά και η ονομασία του ακριβότερου αγορασμένου είδους
18         if COST > MAX_COST:
19             MAX_COST = COST
20             MAX_EIDOS = EIDOS
21     else:
22         print 'Δεν επαρκεί το υπόλοιπο'
23
24     # Εισαγωγή ονομασίας επόμενου είδους προς αγορά
25     EIDOS = raw_input('Δώσε την ονομασία είδους ή TELOS για τερματισμό των αγορών:')
26
27 # 4.3
28 print 'Ακριβότερο προϊόν που αγοράστηκε: ', MAX_EIDOS
29 print 'Το κόστος αγοράς του:', MAX_COST, ' ευρώ'
30 if YPOL>0: # 4.4
31     print 'Υπόλοιπο για άλλες αγορές: ', YPOL, ' ευρώ'
32 else:
33     print 'Μηδενικό υπόλοιπο'
```

ΘΕΜΑ 2

2.1. Στην παρακάτω εντολή σε γλώσσα προγραμματισμού Python, οι μεταβλητές x, y και z είναι τύπου integer και έχουν τις ακόλουθες τιμές: x = -3, y = 2 και z = 4.

b = (9 == 3 * pow(y, 2) - 1) and (abs(x) == z - 1)

Να γράψετε την τιμή που θα πάρει η μεταβλητή b όταν εκτελεστεί η εντολή αυτή.

Μονάδες 5

2.2. Να γράψετε τη λίστα αριθμών που παράγει η καθεμία από τις παρακάτω συναρτήσεις range:

- A. range(5)
- B. range(5, 10)
- Γ. range(5, -1, -1)
- Δ. range(0, 300, 50),

Μονάδες 8

2.3 Δίνεται το παρακάτω πρόγραμμα σε γλώσσα προγραμματισμού Python:

```
1  def add( a, b):  
2      a = a + 5  
3      b = b + 3  
4      print a , b  
5      return a  
6  
7  a = 3  
8  b = 2  
9  result = add( a, b)  
10 print result  
11 print a
```

Να γράψετε τις τιμές που θα εμφανιστούν στην οθόνη κατά την εκτέλεση των εντολών του παραπάνω προγράμματος.

Μονάδες 12

ΘΕΜΑ 2

2.1 Η μεταβλητή *b* θα πάρει την τιμή **False**.

2.2 Οι δοσμένες συναρτήσεις *range* παράγουν τις ακόλουθες λίστες αριθμών:

A. [0, 1, 2, 3, 4]

B. [5, 6, 7, 8, 9]

Γ. [5, 4, 3, 2, 1, 0]

Δ. [0, 50, 100, 150, 200, 250]

2.3 Κατά την εκτέλεση του δοσμένου προγράμματος Python, θα εμφανιστούν στην οθόνη τα ακόλουθα:

8 , 5

8

3

Εξήγηση: Οι παράμετροι *a, b* αλλάζουν τιμές εντός της συνάρτησης (η μεταβλητή *a* παίρνει την τιμή 8 και η *b* την τιμή 5). Αυτές οι τιμές εμφανίζονται με την εντολή ***print a, b*** εντός της συνάρτησης. Στη συνέχεια, επιστρέφεται από τη συνάρτηση, μέσω της εντολής ***return a***, η τιμή 8, η οποία εκχωρείται στη μεταβλητή *result* του κυρίως προγράμματος και εμφανίζεται μέσω της εντολής ***print result***.

Η μεταβίβαση παραμέτρων στην Python λειτουργεί με τέτοιο τρόπο, ώστε οποιαδήποτε αλλαγή στις παραμέτρους εντός της συνάρτησης δεν έχει καμία επίδραση στα ορίσματα-μεταβλητές που έχουν οριστεί εκτός της συνάρτησης (σελ. 79 σχολικού βιβλίου). Έτσι, ενώ η παράμετρος *a* αλλάζει τιμή εντός της συνάρτησης, αυτή η αλλαγή δεν επιστρέφει στο κυρίως πρόγραμμα και η εντολή ***print a*** εμφανίζει την αρχική τιμή της μεταβλητής (δηλ. το 3).

ΘΕΜΑ 4

Η Β' τάξη ενός ΕΠΑΛ πρόκειται να διοργανώσει πολυήμερη εκδρομή στην οποία μπορούν να συμμετάσχουν μέχρι και 50 άτομα. Το κόστος της εκδρομής είναι 250 ευρώ ανά μαθητή, αλλά προσφέρεται έκπτωση 20% ανά παιδί, αν στην εκδρομή συμμετέχουν παιδιά της ίδιας οικογένειας, π.χ. αν στην εκδρομή συμμετέχουν 2 αδέλφια, το καθένα από αυτά δικαιούται έκπτωση 20%.

Να γράψετε πρόγραμμα σε Python το οποίο:

4.1 Για κάθε μαθητή που θα συμμετάσχει στην εκδρομή θα διαβάζει το ονοματεπώνυμο του μαθητή και αν δικαιούται έκπτωση (με έγκυρες τιμές "ΝΑΙ" ή "ΟΧΙ", αντίστοιχα). Η επανάληψη θα τερματίζεται όταν δοθεί σαν όνομα ο χαρακτήρας "-" ή όταν συμπληρωθούν τα 50 άτομα. Θεωρείστε πως όλα τα δεδομένα που εισάγονται είναι έγκυρα.

Μονάδες 10

4.2 Θα υπολογίζει και θα εμφανίζει με κατάλληλο μήνυμα το ονοματεπώνυμο του κάθε μαθητή και το ποσό σε ευρώ που πρέπει να πληρώσει για τη συμμετοχή του στην εκδρομή.

Μονάδες 5

4.3 Θα υπολογίζει και θα εμφανίζει το ποσοστό (%) των μαθητών που δικαιούνται την έκπτωση.

Μονάδες 6

4.4 Εάν τα άτομα που θα συμμετάσχουν στην εκδρομή είναι λιγότερα από 50, θα εμφανίζει το πλήθος τους, αλλιώς θα εμφανίζει το μήνυμα "Συμπληρώθηκε η εκδρομή".

Μονάδες 4

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
    print b
```

ΘΕΜΑ 4

```
1  pl = 0 # Αρχικοποίηση πλήθους συμμετεχόντων μαθητών
2  pl_ekpt = 0 # Αρχικοποίηση πλήθους μαθητών που πήραν έκπτωση
3
4  # Εισαγωγή ονοματεπώνυμου 1ου μαθητή
5  onoma = raw_input('Δώσε το ονοματεπώνυμο του 1ου μαθητή:')
6
7  # Ερώτημα 4.1: Η επανάληψη συνεχίζεται όσο δεν έχει
8  # συμπληρωθεί ο μέγιστος αριθμός μαθητών (50)
9  # και δεν έχει δοθεί ως ονοματεπώνυμο ο χαρακτήρας "-"
10 while pl < 50 and onoma != '-':
11     ekpt= raw_input('Δικαιούται ο μαθητής έκπτωση; (ΝΑΙ/ΟΧΙ):')
12
13     pl = pl +1 # αύξηση του πλήθους των συμμετεχόντων κατά 1
14
15     # 4.2: υπολογισμός του ποσού που πρέπει να πληρώσει ο μαθητής
16     if ekpt == 'ΝΑΙ': # αν δικαιούται έκπτωση 20%
17         poso = 250 * 0.8 # το τελικό ποσό είναι το 80% του αρχικού
18         # 4.3 ενημέρωση του πλήθους των μαθητών που πήραν έκπτωση
19         pl_ekpt = pl_ekpt +1
20     else:
21         poso = 250
22     # 4.2
23     print 'Ο/Η ', onoma, 'πρέπει να πληρώσει για την εκδρομή ', poso, ' ευρώ.'
24
25     # Εισαγωγή ονοματεπώνυμου επόμενου μαθητή ή - για τερματισμό
26     onoma = raw_input('Δώσε ονοματεπώνυμο επόμενου μαθητή ή - για τερματισμό:')
27
28     # 4.3 Ποσοστό % των μαθητών που πήραν έκπτωση
29     pos_ekpt= 100* float(pl_ekpt)/ pl
30     print 'Ποσοστό μαθητών που πήραν έκπτωση: ', pos_ekpt, '%'
31     # Ερώτημα 4.4
32     if pl != 50:
33         print 'Πλήθος συμμετεχόντων μαθητών: ', pl
34     else:
35         print 'Συμπληρώθηκε η εκδρομή'
```

ΘΕΜΑ 2

2.1 Δίνεται το παρακάτω πρόγραμμα σε γλώσσα προγραμματισμού Python:

```
1  def string_func(lexi):
2      if len(lexi) < 2:
3          return '-'
4      return lexi[0:2] + lexi[-2: ]
5
6  word= 'python'
7  print string_func(word)
8  print string_func(word[0])
```

Να γράψετε τις τιμές που θα εμφανιστούν στην οθόνη κατά την εκτέλεση των εντολών του παραπάνω προγράμματος.

Μονάδες 10

2.2 Το ακόλουθο πρόγραμμα σε Python έχει γραφεί για να διαβάζει έναν θετικό ακέραιο και να υπολογίζει και να εμφανίζει το άθροισμα των τετραγώνων των αριθμών από το 1 μέχρι τον θετικό ακέραιο που δόθηκε. Αν, για παράδειγμα, εισαχθεί ο αριθμός 4, το πρόγραμμα θα υπολογίσει και θα εμφανίσει το άθροισμα $1^2 + 2^2 + 3^2 + 4^2$. Στο πρόγραμμα, όμως, υπάρχουν πέντε (5) λάθη. Για κάθε ένα από τα λάθη να γράψετε στο γραπτό σας τον αριθμό της γραμμής στην οποία το εντοπίσατε, να περιγράψετε το λάθος και να χαρακτηρίσετε το λάθος ως συντακτικό ή λογικό. Δεν υπάρχουν περισσότερα του ενός λάθη στην ίδια γραμμή.

```
1  sum=0
2  n=int(input("Δώσε ένα θετικό ακέραιο:"))
3  while n > 0:
4      print 'Ο αριθμός που έδωσες δεν ήταν θετικός!'
5      n=int(input("Δώσε νέο αριθμό:"))
6  for x in range(n):
7      tetr= x ^ 2
8      sum= tetr
9  print "ΑΘΡΟΙΣΜΑ ΤΕΤΡΑΓΩΝΩΝ="  sum
```

Μονάδες 15

ΘΕΜΑ 2

2.1 Κατά την εκτέλεση των εντολών του προγράμματος θα εμφανιστούν στην οθόνη οι ακόλουθες τιμές:

pyon

-

2.2

Αριθμός γραμμής	Περιγραφή λάθους	Είδος λάθους
3	Η ανισότητα στη συνθήκη $n > 0$ είναι λανθασμένη. Για να γίνει σωστά ο έλεγχος εγκυρότητας, η συνθήκη θα έπρεπε να γραφεί: $n \leq 0$	Λογικό
6	<code>range(n)</code> . Υπολογίζεται το άθροισμα των τετραγώνων από το 1 μέχρι το $n-1$. Σωστό: <code>range(n+1)</code>	Λογικό
7	Ο τελεστής που δίνει το τετράγωνο ενός αριθμού γράφεται <code>**</code> (κι όχι <code>^</code>)	Συντακτικό
8	<code>sum= tetr</code> . Λανθασμένος υπολογισμός αθροίσματος. Σωστή εντολή: <code>sum= sum + tetr</code>	Λογικό
9	Λείπει το κόμμα. Σωστή εντολή: <code>print "ΑΘΡΟΙΣΜΑ ΤΕΤΡΑΓΩΝΩΝ=", sum</code>	Συντακτικό

ΘΕΜΑ 4

Μία γκαλερί εμπορεύεται έργα τέχνης που περιλαμβάνουν πίνακες ζωγραφικής και γλυπτά. Να γραφεί πρόγραμμα σε γλώσσα προγραμματισμού Python που:

4.1 Για κάθε έργο τέχνης που διαθέτει η γκαλερί διαβάζει τον τίτλο του, την τιμή πώλησής του σε ευρώ και το είδος του (‘Π’ αν πρόκειται για πίνακα ή ‘Γ’ αν πρόκειται για γλυπτό). Να ελέγχεται η τιμή που εισάγεται για το είδος ως προς την ορθότητά της. Στην περίπτωση λάθους να επανεισάγεται μέχρι να δοθεί ορθή τιμή. Η εισαγωγή των δεδομένων ολοκληρώνεται όταν δίνεται ως τίτλος του έργου η λέξη “ΤΕΛΟΣ”.

Μονάδες 9

4.2 Υπολογίζει και εμφανίζει τον τίτλο του ακριβότερου πίνακα ζωγραφικής, καθώς και την τιμή πώλησής του. Θεωρούμε πως ο πίνακας αυτός είναι μοναδικός.

Μονάδες 8

4.3 Υπολογίζει κι εμφανίζει το πλήθος των γλυπτών που διαθέτει η γκαλερί, καθώς και το συνολικό ποσό σε ευρώ που θα εισέπραττε η γκαλερί αν πωλούνταν όλα τα γλυπτά της.

Μονάδες 8

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1  MAX_TIMH = -1 # 4.2 Αρχικοποίηση της μέγιστης τιμής πίνακα ζωγραφικής
2  pl_glypta = 0 # 4.3 Αρχικοποίηση του πλήθους των γλυπτών
3  sum_glypta = 0 # 4.3 Αρχικοποίηση συνολικού κόστους γλυπτών
4
5  # Εισαγωγή τίτλου 1ου έργου τέχνης
6  TITLE = raw_input('Δώσε τον τίτλο του 1ου έργου τέχνης:')
7
8  # 4.1: η εισαγωγή των δεδομένων συνεχίζεται όσο
9  # δεν έχει δοθεί ως τίτλος έργου τέχνης η λέξη "ΤΕΛΟΣ"
10 while TITLE != 'ΤΕΛΟΣ':
11     TIMH = float(input('Δώσε την τιμή του έργου τέχνης σε ευρώ:'))
12     EIDOS= raw_input('Δώσε το είδος του έργου τέχνης (Π για πίνακα ή Γ για γλυπτό:'))
13     # 4.1 έλεγχος εγκυρότητας των δεδομένων που εισάγονται στο είδος του έργου τέχνης
14     while EIDOS != 'Π' and EIDOS != 'Γ':
15         print 'Άκυρη τιμή !'
16         EIDOS= raw_input('Δώσε Π για πίνακα ή Γ για γλυπτό:')
17     # 4.3: αν πρόκειται για γλυπτό
18     if EIDOS == 'Γ':
19         pl_glypta = pl_glypta + 1
20         sum_glypta = sum_glypta + TIMH
21     else:
22     # 4.2: αν πρόκειται για πίνακα και η τιμή του είναι μεγαλύτερη
23     # από το τρέχον μέγιστο, ενημερώνεται και η μέγιστη τιμή,
24     # αλλά και ο τίτλος του ακριβότερου πίνακα ζωγραφικής
25     if TIMH > MAX_TIMH:
26         MAX_TIMH = TIMH
27         MAX_TITLE = TITLE
28
29     # Εισαγωγή τίτλου επόμενου έργου τέχνης
30     TITLE = raw_input('Δώσε τον τίτλο του έργου τέχνης ή ΤΕΛΟΣ για τερματισμό:')
31
32 # 4.2
33 print 'Ακριβότερος πίνακας ζωγραφικής: ', MAX_TITLE
34 print 'Η τιμή πώλησής του :', MAX_TIMH, ' ευρώ'
35 # 4.3
36 print 'Πλήθος των γλυπτών που διαθέτει η γκαλερί: ', pl_glypta
37 print 'Συνολικό ποσό είσπραξης αν πωλούνταν όλα τα γλυπτά:',sum_glypta, ' ευρώ'
```

ΘΕΜΑ 2

2.1 Δίνεται το παρακάτω πρόγραμμα σε γλώσσα προγραμματισμού Python:

```
1 def example(x):  
2     x= x + '2'  
3     x= x * 2  
4     return x  
5  
6 print example("Hello")
```

A. Αν προσπαθήσουμε να το εκτελέσουμε θα εμφανιστεί κάποιο μήνυμα λάθους ή θα εκτελεστεί κανονικά;

Μονάδες 4

B. Αν στο ερώτημα A απαντήσατε πως θα εμφανιστεί μήνυμα λάθους, να εξηγήσετε γιατί.

Αν απαντήσατε πως θα εκτελεστεί κανονικά, να γράψετε τι θα εμφανιστεί στην οθόνη κατά την εκτέλεση των εντολών του.

Μονάδες 6

2.2. Δίνεται το παρακάτω πρόγραμμα σε γλώσσα προγραμματισμού Python:

```
1 sum=0  
2 for x in range(5, 25, 5):  
3     if x % 5 == 0 and x % 3 == 0:  
4         sum = sum + x  
5     print x  
6 print sum
```

A. Να γράψετε ισοδύναμο πρόγραμμα σε Python, χρησιμοποιώντας την εντολή επανάληψης while αντί της εντολής επανάληψης for.

Μονάδες 8

B. Ποιες τιμές παίρνει η μεταβλητή x κατά την εκτέλεση του προγράμματος;

Μονάδες 4

Γ. Πόσες φορές θα εκτελεστεί η εντολή 5 (**print x**);

Μονάδες 3

ΘΕΜΑ 2

2.1

A. Το πρόγραμμα θα εκτελεστεί κανονικά.

B. Κατά την εκτέλεση του προγράμματος θα εμφανιστεί την οθόνη:

Hello2Hello2

2.2

A.

```
1  sum = 0
2  x = 5
3  while x < 25:
4      if x % 5 == 0 and x % 3 == 0:
5          sum = sum + x
6          print x
7      x = x + 5
8  print sum
```

B. Η μεταβλητή x παίρνει τις εξής τιμές κατά την εκτέλεση του προγράμματος:

5, 10, 15, 20

Γ. Η εντολή 5 (**print x**) θα εκτελεστεί **μία φορά**.

ΘΕΜΑ 4

Να αναπτύξετε πρόγραμμα στην γλώσσα προγραμματισμού Python, το οποίο:

4.1. Να δέχεται 20 ακέραιους αριθμούς στο διάστημα [1,10]. Στην περίπτωση που δοθεί λανθασμένος αριθμός να εμφανίζεται το μήνυμα «Άκυρος αριθμός» και το πρόγραμμα να ζητά να δοθεί ξανά ο αριθμός, μέχρι να δοθεί έγκυρη τιμή.

Μονάδες 6

4.2. Να υπολογίζει και να εμφανίζει το άθροισμα εκείνων των έγκυρων αριθμών που εισήχθησαν ενώ ο προηγούμενος αριθμός που είχε εισαχθεί ήταν ο αριθμός 5. Για παράδειγμα, αν εισαχθούν διαδοχικά οι έγκυροι αριθμοί 4, 2, 6, 1, 1, 3, 5, 10, 4, 5, 2, 7, 8, 10, 9, 4, 3, 1, 3, 3 το ζητούμενο άθροισμα είναι ο αριθμός 12 (συναντάμε δύο φορές το 5 και ακολουθούν οι αριθμοί 10 και 2, αντίστοιχα).

Μονάδες 8

4.3. Να υπολογίζει και να εμφανίζει το ποσοστό (%) των άρτιων αριθμών (στο σύνολο των 20 αριθμών που δόθηκαν).

Μονάδες 6

4.4. Να υπολογίζει και να εμφανίζει το πλήθος των λανθασμένων καταχωρήσεων αριθμών που έγιναν κατά την εισαγωγή των 20 ακεραίων.

Μονάδες 5

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1  sum = 0 # 4.2
2
3  previous=0 # 4.2 Η previous θα κρατάει τον αμέσως προηγούμενο αριθμό
4  pl_art= 0 # 4.3 πλήθος αρτίων
5  pl= 0 # 4.4 πλήθος λανθασμένων καταχωρήσεων αριθμών
6
7  # 4.1 Διαβάζουμε 20 ακεραίους
8  for x in range(20):
9      # 4.1
10     num=int(input("Δώσε ακέραιο από το 1 έως και το 10:"))
11     # έλεγχος εγκυρότητας για τον κάθε αριθμό
12     while num <1 or num>10:
13         print "Ακυρος αριθμός"
14         pl = pl +1 # 4.4 αύξηση των λανθασμένων καταχωρήσεων κατά 1
15         num=int(input("Δώσε ξανά αριθμό (1-10):"))
16
17     # 4.2 Εξετάζουμε αν ο προηγούμενος αριθμός ήταν το 5
18     if previous == 5:
19         sum = sum + num
20
21     # 4.3 Εξετάζουμε αν ο αριθμός είναι άρτιος
22     if num % 2 == 0:
23         pl_art = pl_art +1
24
25     # 4.2 πριν διαβαστεί ο επόμενος αριθμός,
26     # ο τρέχων αριθμός αποθηκεύεται στη μεταβλητή previous
27     previous = num
28
29     # 4.2
30     print "Το άθροισμα των αριθμών που εισήχθησαν μετά από ένα 5:", sum
31     # 4.3
32     print "Το ποσοστό των άρτιων (στο σύνολο των 20 αριθμών):", 100.* pl_art/20, "%"
33     # 4.4
34     print "Πλήθος λανθασμένων καταχωρήσεων αριθμών:", pl
```

ΘΕΜΑ 2

2.1 Δίνεται το παρακάτω πρόγραμμα σε γλώσσα προγραμματισμού Python:

```
1  numList= [0,1,2,3,4,5]
2  for x in range(1,5):
3      numList[x-1]= numList[x]
4
5  for x in range(0,5):
6      print numList[x]
```

Να γράψετε ποιες τιμές θα εμφανιστούν στην οθόνη κατά την εκτέλεση των εντολών του.

Μονάδες 10

2.2 Το ακόλουθο πρόγραμμα σε Python έχει γραφεί για να διαβάζει ακεραίους και να τους τοποθετεί στη λίστα `myList`. Η εισαγωγή ακεραίων τερματίζεται όταν δοθεί το 0 (το 0 δεν εισάγεται στη λίστα). Στη συνέχεια, το πρόγραμμα βρίσκει κι εμφανίζει το μέγιστο από τα στοιχεία της `myList`. Στο πρόγραμμα, όμως, υπάρχουν πέντε (5) λάθη. Για κάθε ένα από τα λάθη να γράψετε στο γραπτό σας τον αριθμό της γραμμής στην οποία το εντοπίσατε, να περιγράψετε το λάθος και να δηλώσετε το είδος του λάθους (λογικό ή συντακτικό):

```
1  myList=[]
2  num=int(input("Δώσε ακέραιο διαφορετικό του 0:"))
3  while num == 0:
4      myList.append(num) # προστίθενται στη λίστα οι αριθμοί μέχρι να δοθεί το 0
5      num=int(input("Δώσε τον επόμενο ακέραιο ή 0 για τερματισμό:"))
6
7  MAX = myList[1]      # θέτουμε ως MAX το αρχικό στοιχείο της λίστας
8  list_length= length(myList)
9  #τα υπόλοιπα στοιχεία της λίστας τα συγκρίνουμε ένα- ένα με το τρέχον MAX
10 for x in range( 1, list_length ):
11     If myList[x] > MAX: # αν το τρέχον στοιχείο είναι μεγαλύτερο του MAX
12         myList[x]= MAX
13
14 print "Ο μέγιστος αριθμός στη λίστα είναι ο " MAX
```

Μονάδες 15

ΘΕΜΑ 2

2.1 Κατά την εκτέλεση των εντολών του θα εμφανιστούν στην οθόνη οι εξής τιμές:

1

2

3

4

4

2.2

Αριθμός γραμμής	Περιγραφή λάθους	Είδος λάθους
3	Εισάγουμε τους αριθμούς στη λίστα εφόσον είναι διαφορετικοί του 0. Επομένως στη εντολή while num == 0 ο τελεστής == θα έπρεπε να είναι !=	Λογικό
7	Πρέπει να θέσουμε στο MAX το αρχικό στοιχείο της λίστας κι επομένως στην εντολή MAX = myList[1] , το 1 είναι λάθος (το myList[1] είναι το 2 ^ο στοιχείο της λίστας). <u>Σωστό:</u> MAX= myList[0]	Λογικό
8	Το μήκος της λίστας δίνεται από τη συνάρτηση len() κι όχι length() . <u>Σωστή εντολή:</u> list_length = len(myList)	Συντακτικό
12	Αν το τρέχον στοιχείο της λίστας είναι μεγαλύτερο από το MAX, τότε στο MAX εκχωρείται το τρέχον στοιχείο της λίστας κι όχι το ανάποδο. Η εντολή myList[x]= MAX πρέπει να αλλάξει σε MAX = myList[x]	Λογικό
14	Λείπει το κόμμα (',') ανάμεσα στο μήνυμα και τη μεταβλητή MAX	Συντακτικό

ΘΕΜΑ 4

Μια επαγγελματική ομάδα καλαθοσφαίρισης (μπάσκετ) πρέπει να έχει στο δυναμικό της από 12 μέχρι 15 επαγγελματίες παίκτες, Έλληνες κι αλλοδαπούς. Να αναπτύξετε πρόγραμμα στη γλώσσα προγραμματισμού Python, το οποίο:

4.1. Να διαβάζει, αρχικά, το πλήθος των παικτών της ομάδας, ελέγχοντας ότι είναι μέσα στα επιτρεπτά όρια. Στην περίπτωση που εισάγεται άκυρη τιμή, το πρόγραμμα πρέπει να ζητάει νέα τιμή μέχρι να δοθεί τιμή εντός ορίων.

Μονάδες 5

4.2 Στη συνέχεια, να ζητά από τον χρήστη το όνομα κάθε παίκτη, την προέλευσή του ('Ε' για Έλληνα, 'Α' για αλλοδαπό), την ηλικία του και το ύψος του σε μέτρα. Να θεωρήσετε ότι τα δεδομένα δίνονται σωστά και δεν χρειάζεται οποιοσδήποτε έλεγχος.

Μονάδες 5

4.3. Να βρίσκει και να εμφανίζει το όνομα και το ύψος του κοντούτερου παίκτη της ομάδας (να θεωρήσετε ότι ένας παίκτης είναι πιο κοντός από τους υπόλοιπους).

Μονάδες 7

4.4. Να υπολογίζει και να εμφανίζει για το σύνολο των Ελλήνων παικτών, ποιο είναι το ποσοστό (%) εκείνων που έχουν ηλικία μικρότερη των 20 χρόνων. Αν, για παράδειγμα, οι Έλληνες παίκτες είναι 10, εκ των οποίων οι 4 έχουν ηλικία μικρότερη των 20 ετών, το ποσοστό που πρέπει να εμφανιστεί είναι 40%. Θεωρείστε ότι υπάρχει στην ομάδα τουλάχιστον ένας Έλληνας παίκτης.

Μονάδες 8

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1 min_height = 10 # 4.3 Αρχικοποίηση της τιμής του ελάχιστου ύψους
2 ntopioi = 0 # 4.4 Αρχικοποίηση του πλήθους των Ελλήνων παικτών
3 young = 0 # 4.4 Αρχικοποίηση του πλήθους των Ελλήνων με ηλικία < 20
4
5 # Ερώτημα 4.1: Εισαγωγή πλήθους παικτών
6 num = int(input("Δώσε το πλήθος των παικτών της ομάδας (12-15):"))
7 while num < 12 or num > 15: # 4.1 έλεγχος εγκυρότητας για το πλήθος των παικτών
8     print "Άκυρο πλήθος παικτών !"
9     num = int(input("Δώσε ξανά το πλήθος των παικτών της ομάδας (12-15):"))
10
11 # Ερώτημα 4.2: Εισαγωγή δεδομένων για τους παίκτες της ομάδας
12 for x in range(num):
13     name = raw_input("Δώσε το ονοματεπώνυμο του παίκτη:")
14     origin=raw_input("Δώσε την προέλευσή του ('Ε' για Έλληνα, 'Α' για αλλοδαπό):")
15     age = int(input("Δώσε την ηλικία του παίκτη:"))
16     height = float(input("Δώσε το ύψος του (σε μέτρα):"))
17
18 # Ερώτημα 4.3: Αν το ύψος του παίκτη είναι μικρότερο από το τρέχον ελάχιστο
19 # ενημερώνεται και το ελάχιστο ύψος, αλλά και το όνομα του κοντύτερου παίκτη
20 if height < min_height:
21     min_height = height
22     shortest = name
23
24 # Ερώτημα 4.4:
25 if origin == 'Ε' : # αν ο παίκτης είναι Έλληνας
26     ntopioi = ntopioi + 1 # αυξάνουμε το πλήθος των Ελλήνων κατά 1
27     if age < 20: # κι εξετάζουμε αν έχει ηλικία μικρότερη των 20 ετών
28         young = young +1
29
30 # Ερώτημα 4.3:
31 print "Κοντύτερος παίκτης της ομάδας: ", shortest
32 print "Με ύψος: ", min_height, " μέτρα."
33 # Ερώτημα 4.4:
34 pos =100. * young / float(ntopioi)
35 print "Το ποσοστό εκείνων με ηλικία <20 (στο σύνολο των Ελλήνων παικτών): ", pos, "%"
```

ΘΕΜΑ 2

2.1.

Να μετατρέψετε τις πιο κάτω λεκτικές προτάσεις στις αντίστοιχες λογικές εκφράσεις στη γλώσσα προγραμματισμού Python, χωρίς να κάνετε χρήση συναρτήσεων:

1. Η τιμή της μεταβλητής number να είναι περιττός αριθμός ή μεγαλύτερος του 50.
2. Η τιμή της μεταβλητής bathmos να είναι μεταξύ του 1 και του 20 συμπεριλαμβανομένων.
3. Η τιμή της μεταβλητής poso να είναι πολλαπλάσιο του 5 και να μην ισούται με 100.
4. Το τετράγωνο της μεταβλητής x μειωμένο κατά 10 να είναι αρνητικός αριθμός.
5. Το πηλίκο της διαίρεσης της μεταβλητής d με το 2 να ισούται με τον αριθμό 3.5

Μονάδες 15

2.2

A. Να γράψετε ισοδύναμο με το παρακάτω τμήμα προγράμματος Python, χρησιμοποιώντας τη δομή επανάληψης for αντί της δομής επανάληψης while.

```
1  x = 0
2  k = 5
3  while k > 0:
4      y = k + x
5      print x , y
6      x = x + 1
7      k = k - 1
```

Μονάδες 5

B. Ποιες τιμές θα εμφανιστούν κατά τη διάρκεια εκτέλεσης του παραπάνω τμήματος προγράμματος;

Μονάδες 5

ΘΕΜΑ 2

2.1

1. $(\text{number} \% 2 == 1) \text{ or } (\text{number} > 50)$
2. $(\text{bathmos} \geq 1) \text{ and } (\text{bathmos} \leq 20)$
3. $(\text{poso} \% 5 == 0) \text{ and } (\text{poso} \neq 100)$
4. $x ** 2 - 10 < 0$
5. $d / 2. == 3.5$

2.2

A.

```
1    x = 0
2    for k in range(5, 0, -1):
3        y = k + x
4        print x , y
5        x = x + 1
```

B. Κατά τη διάρκεια εκτέλεσης του παραπάνω τμήματος προγράμματος θα εμφανιστούν οι εξής τιμές:

```
0 5
1 5
2 5
3 5
4 5
```


ΘΕΜΑ 4

Το Μουσείο Φυσικής Ιστορίας της περιοχής σας προσφέρει 5 εκπαιδευτικά προγράμματα για παιδιά. Το εισιτήριο για κάθε παιδί πωλείται προς 7 ευρώ. Το Μουσείο καταγράφει τα εισιτήρια που πουλήθηκαν τον περασμένο μήνα ανά πρόγραμμα, με σκοπό να αποφασίσει τη συνέχιση ή μη του κάθε προγράμματος. Κριτήριο για τη συνέχιση κάθε προγράμματος είναι να έχει φέρει έσοδα στο Μουσείο τουλάχιστον 1500 ευρώ κατά τον περασμένο μήνα.

Να γραφεί πρόγραμμα σε Python που:

4.1 Για κάθε εκπαιδευτικό πρόγραμμα διαβάζει τον τίτλο του προγράμματος. Διαβάζει, επίσης το πλήθος των εισιτηρίων που πουλήθηκαν για το πρόγραμμα αυτό τον περασμένο μήνα, ελέγχοντας ότι ο αριθμός που καταχωρείται είναι μεγαλύτερος ή ίσος του μηδενός.

Μονάδες 6

4.2 Το πρόγραμμα υπολογίζει τα μηνιαία έσοδα από τις πωλήσεις εισιτηρίων του προγράμματος αυτού. Εμφανίζει, ακολούθως, τον τίτλο του προγράμματος, τα μηνιαία έσοδά του και κατάλληλο μήνυμα για τη συνέχιση ή μη του προγράμματος.

Μονάδες 7

4.3 Υπολογίζει κι εμφανίζει το πλήθος των προγραμμάτων που πούλησαν περισσότερα από 1000 εισιτήρια το καθένα.

Μονάδες 7

4.4 Υπολογίζει κι εμφανίζει το συνολικό ποσό σε ευρώ που εισέπραξε το Μουσείο τον περασμένο μήνα από τις πωλήσεις εισιτηρίων και των 5 εκπαιδευτικών προγραμμάτων.

Μονάδες 5

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1  sum = 0.0 # 4.4 Αρχικοποίηση συνολ. ποσού από τα εισιτήρια και των 5 προγραμμάτων
2  pl=0 # 4.3 Αρχικοποίηση πλήθους προγραμμάτων που πούλησαν >1000 εισιτήρια
3
4  # Ερώτημα 4.1: Εισαγωγή δεδομένων για το κάθε εκπαιδευτικό πρόγραμμα
5  for x in range(5):
6      title = raw_input("Δώσε τον τίτλο του προγράμματος:")
7      tickets = int(input("Πλήθος εισιτηρίων που πούλησε το πρόγραμμα τον περασμένο μήνα:"))
8      while tickets < 0:
9          print "Τα εισιτήρια δεν μπορεί να είναι αρνητικός αριθμός!"
10         tickets = int(input("Δώσε ξανά τα εισιτήρια που πούλησε τον περασμένο μήνα:"))
11
12 # Ερώτημα 4.2
13     esoda= tickets * 7 # Υπολογισμός των εσόδων από τα εισιτήρια του προγράμματος
14     print title, esoda # εμφάνιση τίτλου και εσόδων προγράμματος
15     # εμφάνιση μηνύματος σχετικού με τη συνέχιση ή μη του προγράμματος
16     if esoda >= 1500:
17         print "Το πρόγραμμα θα συνεχιστεί"
18     else:
19         print "Το πρόγραμμα δεν θα συνεχιστεί"
20
21 # 4.3 έλεγχος αν το πρόγραμμα πούλησε >1000 εισιτήρια
22     if tickets > 1000:
23         pl= pl +1
24
25     sum = sum + esoda # 4.4 Πρόσθεση των εσόδων του προγράμματος στο σύνολο
26 # 4.3:
27     print "Πλήθος προγραμμάτων που το καθένα πούλησε περισσότερα από 1000 εισιτήρια: ", pl
28 # 4.4:
29     print "Τα συνολικά έσοδα (σε ευρώ) και από τα 5 προγράμματα μαζί είναι: ", sum
```

ΘΕΜΑ 2

2.1. Δίνεται το παρακάτω πρόγραμμα σε γλώσσα προγραμματισμού Python:

```
1  x = "τέλεια"  
2  
3  def myfunc( x ):  
4      x = "φανταστική"  
5      return x  
6  
7  message= "Η Python είναι " + myfunc( x )  
8  print message  
9  print "Είναι πραγματικά " + x
```

Να γράψετε τι θα εμφανιστεί στην οθόνη κατά την εκτέλεση των εντολών του παραπάνω προγράμματος.

Μονάδες 10

2.2 Ποιες από τις παρακάτω εντολές είναι έγκυρες στη γλώσσα προγραμματισμού Python:

A. test= "Python" * 4 *2

B. x ← 40

Γ. x , y = 50 , 100

Δ. num = float("3") + 2.5

Ε. 1a= "33"

Για εκείνες τις εντολές που είναι λανθασμένες, να περιγράψετε το λάθος τους.

Μονάδες 15

ΘΕΜΑ 2

2.1 Κατά την εκτέλεση του δοσμένου προγράμματος Python, θα εμφανιστούν στην οθόνη τα ακόλουθα:

Η Python είναι φανταστική

Είναι πραγματικά τέλεια

Εξήγηση: Εντός της συνάρτησης myfunc(), η παράμετρος x παίρνει την τιμή "φανταστική". Η τιμή αυτή επιστρέφεται από τη συνάρτηση στο κυρίως πρόγραμμα, μέσω της εντολής **return x**, συνενώνεται με τη συμβολοσειρά "Η Python είναι " και το αποτέλεσμα της συνένωσης εκχωρείται στη μεταβλητή message (στη γραμμή 7 του προγράμματος). Επομένως, η εντολή **print message** εμφανίζει στην οθόνη: **Η Python είναι φανταστική.**

Η μεταβίβαση παραμέτρων στην Python λειτουργεί με τέτοιο τρόπο, ώστε οποιαδήποτε αλλαγή στις παραμέτρους εντός της συνάρτησης δεν έχει καμία επίδραση στα ορίσματα-μεταβλητές που έχουν οριστεί εκτός της συνάρτησης (σελ. 79 σχολικού βιβλίου). Έτσι, ενώ η παράμετρος x αλλάζει τιμή εντός της συνάρτησης, αυτή η αλλαγή δεν επιστρέφει στο κυρίως πρόγραμμα και η εντολή **print "Είναι πραγματικά " + x** εμφανίζει στην οθόνη:

Είναι πραγματικά τέλεια.

2.2 Από τις εντολές που δίνονται, είναι έγκυρες στη γλώσσα προγραμματισμού Python οι εξής:

A, Γ, Δ

- Η εντολή B ($x \leftarrow 40$) είναι λανθασμένη γιατί στην Python η εκχώρηση τιμής σε μια μεταβλητή γίνεται με το ίσον (=) κι όχι με το βελάκι ($\leftarrow$). (σελ. 45 σχολικού βιβλίου)
- Η εντολή E (1a= "33") είναι λανθασμένη γιατί στην Python δεν επιτρέπεται τα ονόματα των μεταβλητών να ξεκινούν με ψηφίο. (σελ. 45 σχολικού βιβλίου)

ΘΕΜΑ 4

Στις βουλευτικές εκλογές μιας μικρής χώρας συμμετέχουν 9 κόμματα. Ο κάθε πολίτης ψηφίζει υπέρ ενός μόνο κόμματος και δεν υπάρχουν λευκά ψηφοδέλτια. Για να έχει ένα κόμμα εκπροσώπηση στο κοινοβούλιο της χώρας πρέπει να συγκεντρώσει το 3% τουλάχιστον των συνολικών έγκυρων ψήφων. Να γραφεί πρόγραμμα σε Python που:

4.1 Διαβάζει το συνολικό πλήθος των έγκυρων ψηφοδελτίων. Για κάθε κόμμα, διαβάζει το όνομά του και το πλήθος των ψήφων που συγκέντρωσε στις εκλογές. Θεωρείστε ότι όλα τα δεδομένα που πληκτρολογούνται είναι έγκυρα και δεν απαιτείται κανένας έλεγχος των τιμών που εισάγονται.

Μονάδες 4

4.2

A. Το πρόγραμμα βρίσκει, με τη βοήθεια της συνάρτησης του παρακάτω ερωτήματος B, αν το συγκεκριμένο κόμμα θα μπει στο κοινοβούλιο κι εμφανίζει κατάλληλο μήνυμα.

Μονάδες 5

B. Υλοποιεί συνάρτηση που δέχεται ως παραμέτρους τις ψήφους ενός κόμματος και τις συνολικές έγκυρες ψήφους και επιστρέφει τη λέξη «ΝΑΙ», εφόσον το κόμμα μπαίνει στο κοινοβούλιο ή «ΟΧΙ» στην αντίθετη περίπτωση.

Μονάδες 5

4.3 Εμφανίζει το όνομα του νικητήριου κόμματος των εκλογών, καθώς και το ποσοστό επί τοις εκατό (%) των ψήφων που συγκέντρωσε. Θεωρείστε πως δεν υπήρξε ισοψηφία στις εκλογές.

Μονάδες 7

4.4 Εμφανίζει το πλήθος των κομμάτων που έμειναν εκτός Βουλής.

Μονάδες 4

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
    print b
```

ΘΕΜΑ 4

```
1  # 4.2 B.
2  def sti_vouli( v, v_total):
3      if v >= v_total * 0.03:
4          return "NAI"
5      else:
6          return "OXI"
7
8  # κυρίως πρόγραμμα
9  max_v=-1 # 4.3
10 pl=0 # 4.4
11
12 # Ερώτημα 4.1
13 egkyra=int(input("Δώσε το πλήθος των έγκυρων ψηφοδελτίων:"))
14
15 for i in range(9):
16     komma=raw_input("Δώσε το όνομα του κόμματος:")
17     votes=int(input("Δώσε το πλήθος των ψήφων που συγκέντρωσε το κόμμα αυτό:"))
18
19 # 4.2 A. Βρίσκουμε αν το κόμμα αυτό μπαίνει στη Βουλή
20 apotelesma= sti_vouli(votes,egkyra) # κλήση της συνάρτησης
21 if apotelesma=="NAI" :
22     print "Το κόμμα ", komma , " μπαίνει στη Βουλή!"
23 else:
24     print "Το κόμμα ", komma , " δεν μπαίνει στη Βουλή!"
25     pl= pl+1
26
27 # 4.3 Αν οι ψήφοι αυτού του κόμματος είναι > από το τρέχον μέγιστο (max_v),
28 # ενημερώνονται τόσο οι μέγιστες ψήφοι, όσο και το όνομα του κόμματος
29 # που τις πήρε (winner)
30 if votes > max_v :
31     max_v = votes
32     winner = komma
33
34 # 4.3 Εμφάνιση στοιχείων νικητήριου κόμματος
35 print "Νικητήριο κόμμα:", winner
36 print "Με ποσοστό ψήφων:", float(max_v)*100/ egkyra, "%"
37
38 # 4.4
39 print "Πλήθος κομμάτων που έμειναν εκτός Βουλής:", pl
```

ΘΕΜΑ 2

2.1 Δίνεται το παρακάτω πρόγραμμα σε γλώσσα προγραμματισμού Python:

```
1 def change_s(word):
2     if len(word) < 2:
3         return word
4     return word [-1] + word [1 : len(word)-1] + word [0]
5
6 print change_s('12345')
7 print change_s ('1')
```

Να γράψετε τις τιμές που θα εμφανιστούν στην οθόνη κατά την εκτέλεση των εντολών του παραπάνω προγράμματος.

Μονάδες 10

2.2 Δίνονται τα ακόλουθα δύο προγράμματα σε Python:

A	B
<pre>n = int(input()) if n < 1000: print '*' if n < 100: print '*' if n < 10: print '*' if n < 1: print '*'</pre>	<pre>n = int(input()) if n < 1000: print '*' elif n < 100: print '*' elif n < 10: print '*' elif n < 1: print '*'</pre>

Είναι τα προγράμματα αυτά ισοδύναμα; Δικαιολογείστε την απάντησή σας.

Μονάδες 15

ΘΕΜΑ 2

2.1 Κατά την εκτέλεση του δοσμένου προγράμματος Python, θα εμφανιστούν στην οθόνη τα ακόλουθα:

52341

1

2.2 Τα δύο αυτά προγράμματα δεν είναι ισοδύναμα.

Στο πρόγραμμα A, όλες οι συνθήκες των if θα εξεταστούν διαδοχικά. Αντίθετα, στο πρόγραμμα B αν κάποια από τις συνθήκες των if ή elif βρεθεί Αληθής (True), τότε θα εκτελεστεί η εντολή που την ακολουθεί και οι υπόλοιπες συνθήκες από εκεί και κάτω δεν θα ελεγχθούν. Ο έλεγχος του προγράμματος θα μεταφερθεί στην επόμενη εντολή μετά το τέλος της δομής επιλογής.

Αν, για παράδειγμα, δοθεί σαν είσοδος η τιμή 0, το πρόγραμμα A θα εμφανίσει συνολικά τέσσερα '*', αφού και οι τέσσερις συνθήκες είναι αληθείς (True). Το πρόγραμμα B, αντίθετα, θα εμφανίσει μόνο ένα '*':

Η πρώτη συνθήκη ($n < 1000$) θα βρεθεί αληθής, και θα εκτελεστεί η εντολή `print '*'`.

Οι συνθήκες των elif δεν θα ελεγχθούν.

ΘΕΜΑ 4

Στο πλαίσιο της Γιορτής Κρασιού που έγινε στην Αττική, πραγματοποιήθηκε, όπως κάθε χρόνο, διαγωνισμός στον οποίο συμμετείχαν 60 οινοποιεία απ' όλη την ελληνική επικράτεια. Με βάση τον κανονισμό, κάθε οινοποιείο συμμετέχει στο διαγωνισμό με 3 συγκεκριμένα είδη κρασιών. Νικητής του διαγωνισμού ανακηρύσσεται το οινοποιείο που συγκεντρώνει το μεγαλύτερο συνολικό βαθμό για τα 3 κρασιά του. Ο βαθμός αυτός είναι ο μέσος όρος των βαθμών για τα 3 κρασιά του οινοποιείου. Να αναπτύξετε πρόγραμμα σε Python, το οποίο:

4.1 Για κάθε οινοποιείο διαβάζει την επωνυμία του και τους βαθμούς που πήρε για τα 3 κρασιά του σε λίστα `bathm`. Οι βαθμοί είναι ακέραιοι στο διάστημα $[1,10]$. Να γίνεται έλεγχος των βαθμών που εισάγονται ως προς την εγκυρότητά τους. Στην περίπτωση λάθους να ζητείται επανεισαγωγή των βαθμών μέχρι να δοθούν έγκυρες τιμές.

Μονάδες 12

4.2 Υπολογίζει, με τη βοήθεια συνάρτησης την οποία θα υλοποιήσετε, το συνολικό βαθμό του κάθε οινοποιείου. Η συνάρτηση αυτή θα δέχεται σαν παραμέτρους τους 3 βαθμούς κάθε οινοποιείου και θα επιστρέφει τον μέσο όρο τους. Το κυρίως πρόγραμμα εμφανίζει την επωνυμία του οινοποιείου, ακολουθούμενη από το συνολικό βαθμό του.

Μονάδες 6

4.3 Το πρόγραμμα βρίσκει κι εμφανίζει την επωνυμία του οινοποιείου που κέρδισε στον διαγωνισμό, καθώς και το συνολικό βαθμό του. Θεωρούμε πως δεν υπήρξε ισοβαθμία.

Μονάδες 7

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
    print b
```

ΘΕΜΑ 4

```
1  # 4.2
2  def mo(vath1, vath2, vath3):
3      return (vath1+vath2+ vath3)/3.
4
5  # κυρίως πρόγραμμα
6  max_v=-1 # 4.3 Αρχικοποίηση μέγιστου συνολικού βαθμού
7  # Δημιουργία και αρχικοποίηση της λίστας των 3 βαθμών ενός οινοποιείου
8  bathm= [0,0,0]
9
10 # Ερώτημα 4.1
11 for i in range(60):
12     winery=raw_input("Δώσε την επωνυμία του οινοποιείου:")
13     for k in range(3):
14         print "Βαθμός ", k+1
15         num=int(input("Δώσε τον βαθμό του οινοποιείου (1-10):"))
16         # έλεγχος εγκυρότητας για τον κάθε βαθμό
17         while num <1 or num>10:
18             print "'Ακυρη τιμή !"
19             num=int(input("Δώσε ξανά το βαθμό (1-10):"))
20         bathm[k]=num # βάλε το βαθμό στην αντίστοιχη θέση της λίστας
21
22 # 4.2 Βρίσκουμε κι εμφανίζουμε το συνολικό βαθμό του οινοποιείου
23 total= mo(bathm[0], bathm[1], bathm[2])
24 print "Οινοποιείο: ", winery, ". Συνολικός βαθμός: " , total
25
26 # 4.3 Αν ο συν. βαθμός του οινοποιείου είναι > από το τρέχον μέγιστο,
27 # ενημερώνεται τόσο ο μέγιστος βαθμός, όσο και το όνομα του
28 # οινοποιείου που τον πήρε (νικητής)
29 if total> max_v :
30     max_v= total
31     winner= winery
32
33 # 4.3 Εμφάνιση στοιχείων νικητή
34 print "Οινοποιείο που κέρδισε στο διαγωνισμό:", winner
35 print "Ο συνολικός βαθμός του:", max_v
```

ΘΕΜΑ 2

2.1

Να αντιστοιχίσετε κάθε στοιχείο της στήλης Α του ακόλουθου πίνακα, με το κατάλληλο στοιχείο της στήλης Β. Ένα από τα στοιχεία της στήλης Β περισσεύει:

Στήλη Α	Στήλη Β
1. True	Α. Αριθμητικός τελεστής
2. float	Β. Συγκριτικός τελεστής
3. **	Γ. Λογικός Τελεστής
4. not	Δ. Ενσωματωμένη συνάρτηση
5. !=	Ε. Μη ενσωματωμένη συνάρτηση
	ΣΤ. Λογική σταθερά (τιμή)

Μονάδες 5

2.2 Από τα ακόλουθα ονόματα μεταβλητών να επιλέξετε εκείνα που είναι έγκυρα στη γλώσσα προγραμματισμού Python:

A. integer	B. 2x	Γ. x2	Δ. x-2	Ε. _x2	ΣΤ. x_2	Ζ. x2_	Η. x2_\$
------------	-------	-------	--------	--------	---------	--------	----------

Μονάδες 5

2.3

Το παρακάτω πρόγραμμα σε Python έχει γραφεί για να ζητάει έναν ακέραιο μεγαλύτερο του 3 (με έλεγχο εγκυρότητας του αριθμού που πληκτρολογεί ο χρήστης). Στη συνέχεια, να υπολογίζει και να εμφανίζει το γινόμενο των ακεραίων που είναι πολλαπλάσια του 3, ξεκινώντας από τον αριθμό 3 και φτάνοντας μέχρι και τον ακέραιο num που έδωσε ο χρήστης. Αν, για παράδειγμα, δοθεί ως num ο αριθμός 6, τότε το ζητούμενο γινόμενο είναι 18 (αφού στο διάστημα [3,6] οι αριθμοί 3 και 6 είναι πολλαπλάσια του 3).

Στο πρόγραμμα υπάρχουν πέντε (5) κενά, τα οποία έχουν αριθμηθεί και υπογραμμιστεί. Να γράψετε στο γραπτό σας τον αριθμό του κάθε κενού και δίπλα τι πρέπει να συμπληρωθεί, ώστε το πρόγραμμα να εκτελεί τη λειτουργία που περιγράφηκε.

```
1  gin = ...(1)...
2  num= int(input("Δώσε έναν ακέραιο μεγαλύτερο του 3:"))
3  while num ...(2)... 3 :
4      print "Άκυρος αριθμός!"
5      num= int(input("Δώσε έναν ακέραιο μεγαλύτερο του 3:"))
6  for x in range(3, ...(3)...):
7      if x ...(4)... :
8          gin = ...(5)...
9  print "Γινόμενο:", gin
```

Μονάδες 15

ΘΕΜΑ 2

2.1

1. $\rightarrow \Sigma\Gamma$.

2. $\rightarrow \Delta$.

3. $\rightarrow A$.

4. $\rightarrow \Gamma$.

5. $\rightarrow B$.

2.2 Τα έγκυρα ονόματα μεταβλητών είναι τα εξής: **A, Γ, E, ΣΤ, Z**

2.3

(1): 1

(2): $\leq$

(3): $\text{num} + 1$

(4): $\% 3 == 0$

(5): $\text{gin} * x$ ή $x * \text{gin}$

ΘΕΜΑ 4

Μια εταιρεία παροχής υπηρεσιών κινητής τηλεφωνίας, έκανε αναπροσαρμογή των χρεώσεών της. Η μηνιαία χρέωση χρόνου ομιλίας του κάθε συνδρομητή, υπολογίζεται τώρα κλιμακωτά σύμφωνα με τον πιο κάτω πίνακα:

Μηνιαίος χρόνος συνομιλιών (σε ακέραια λεπτά)	Χρέωση χρόνου ομιλίας (σε ευρώ ανά λεπτό)
1 - 100	0.06
101 - 300	0.05
> 300	0.04

Εάν, για παράδειγμα, ένας συνδρομητής συμπλήρωσε συνολικά σ' ένα μήνα 450 λεπτά τηλεφωνικής συνομιλίας, τότε για τα πρώτα 100 λεπτά θα πληρώσει 0.06 ευρώ ανά λεπτό ομιλίας, για τα επόμενα 200 λεπτά θα πληρώσει 0.05 ευρώ ανά λεπτό ομιλίας και για τα υπόλοιπα 150 λεπτά θα πληρώσει 0.04 ευρώ ανά λεπτό ομιλίας.

Στην παραπάνω χρέωση χρόνου ομιλίας (ΧΡ1) προστίθεται ο ΦΠΑ , που ανέρχεται στο 24% της χρέωσης ΧΡ1, για να προκύψει το τελικό ποσό που επιβαρύνει το συνδρομητή.

Να αναπτύξετε πρόγραμμα στην γλώσσα προγραμματισμού Python, το οποίο:

4.1. Για κάθε συνδρομητή θα διαβάζει τον αριθμό τηλεφώνου, το ονοματεπώνυμο (όνομα και επίθετο σε μια μεταβλητή) και τα ακέραια λεπτά τηλεφωνικής επικοινωνίας του για τον περασμένο μήνα. Να θεωρήσετε ότι τα δεδομένα εισάγονται σωστά και δεν χρειάζεται οποιοσδήποτε έλεγχος εγκυρότητας.

Μονάδες 3

4.2. Θα υπολογίζει το ποσό του μηνιαίου λογαριασμού του συνδρομητή, λαμβάνοντας υπόψη τη χρέωση χρόνου ομιλίας και τον ΦΠΑ, και στη συνέχεια θα εμφανίζει με κατάλληλο μήνυμα το ονοματεπώνυμο, τον αριθμό τηλεφώνου και τη μηνιαία χρέωση του συνδρομητή.

Μονάδες 12

4.3. Η διαδικασία θα επαναλαμβάνεται συνεχώς για διάφορους συνδρομητές και θα τερματίζεται όταν ο χρήστης δώσει αριθμό τηλεφώνου 999999999. Να θεωρήσετε ότι θα εισαχθούν δεδομένα για τουλάχιστον έναν συνδρομητή.

Μονάδες 4

4.4. Το πρόγραμμα θα υπολογίζει και θα εμφανίζει το πλήθος των συνδρομητών των οποίων ο μηνιαίος λογαριασμός ξεπέρασε τα 50 ευρώ.

Μονάδες 6

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1  # 4.4 Αρχικοποίηση του πλήθους των συνδρομητών που χρεώθηκαν >50 ευρώ
2  pl_50 = 0
3
4  # Εισαγωγή τηλεφώνου 1ου συνδρομητή
5  tel = int(input('Δώσε το τηλέφωνο του 1ου συνδρομητή:'))
6
7  # 4.3: η εισαγωγή των δεδομένων συνεχίζεται όσο
8  # δεν έχει δοθεί ως τηλέφωνο το "999999999"
9  while tel != 999999999:
10     onoma= raw_input('Δώσε το ονοματεπώνυμο του συνδρομητή:')
11     xron= int(input('Δώσε τα ακέραια λεπτά χρόνου ομιλίας του συνδρομητή:'))
12     # 4.2 υπολογισμός χρέωσης χρόνου ομιλίας
13     if xron <= 100 :    # τα πρώτα 100 λεπτά ομιλίας προς 0.06/λεπτό
14         XP1= xron * 0.06
15     elif xron <= 300 :  # τα επόμενα 200 λεπτά προς 0.05/λεπτό
16         XP1= 100 * 0.06 + (xron-100) * 0.05
17     else:    # μετά τα 300 λεπτά η χρέωση είναι 0.04/λεπτό
18         XP1= 100 * 0.06 + 200 * 0.05 + (xron-300) * 0.04
19
20     FPA= XP1 * 24/100
21     TEL_XP = XP1 + FPA
22     print "Ο συνδρομητής ", onoma, " με τηλέφωνο ", tel, " οφείλει για το μήνα ", TEL_XP, " ευρώ."
23
24     # 4.4 Εξετάζουμε αν ο λογαριασμός ξεπέρασε τα 50 ευρώ
25     if TEL_XP > 50 :
26         pl_50 = pl_50 +1
27
28     # Εισαγωγή τηλεφώνου επόμενου συνδρομητή ή 999999999 για τερματισμό
29     tel = int(input('Δώσε το τηλέφωνο του επόμενου συνδρομητή ή 999999999 για τερματισμό:'))
30
31 # 4.4
32 print 'Πλήθος συνδρομητών με μηνιαίο λογαριασμό > 50 ευρώ: ', pl_50
```


ΘΕΜΑ 2

2.1. Να γράψετε την παρακάτω αριθμητική παράσταση σαν εντολή εκχώρησης τιμής στη γλώσσα προγραμματισμού Python.

$$y = 4 + x^2 + \frac{2(x+1)}{x^3+1} - |x|$$

Μονάδες 7

2.2. Να χαρακτηρίσετε τα ονόματα των μεταβλητών που ακολουθούν, γράφοντας στο γραπτό σας, δίπλα στο γράμμα που αντιστοιχεί σε κάθε όνομα μεταβλητής τη λέξη **Αποδεκτό**, αν το όνομα της μεταβλητής είναι αποδεκτό στη γλώσσα προγραμματισμού Python ή τις λέξεις **Μη Αποδεκτό**, αν το όνομα της μεταβλητής είναι μη αποδεκτό.

- α. a_
- β. 3poli
- γ. print
- δ. G_1
- ε. FALSE
- στ. EPA.L

Μονάδες 3

2.3. Δίνεται το παρακάτω πρόγραμμα σε γλώσσα Python. Τι πιστεύετε ότι κάνει το πρόγραμμα; Να ξαναγράψετε το πρόγραμμα στο γραπτό σας χωρίς εμφωλευμένα if, χρησιμοποιώντας λογικούς τελεστές και την εντολή elif (πολλαπλή επιλογή).

```
1  m= input(' Δώσε το βάρος σου (σε κιλά): ')
2  h= input(' Δώσε το ύψος σου (σε μέτρα): ')
3  dms=m/h**2
4  if dms>=18.5:
5      # Εμφωλευμένο if
6      if dms <= 24.99:
7          print ' Φυσιολογικό Βάρος '
8      else:
9          if dms <= 29.99:
10             print ' Προ-παχυσαρκία '
11         else:
12             if dms<= 34.99:
13                 print 'Παχυσαρκία τύπου I'
14             else:
15                 if dms<= 39.99:
16                     print 'Παχυσαρκία τύπου II'
17                 else:
18                     print 'Παχυσαρκία τύπου III'
19     else:
20         print 'Ελλιποβαρής'
```

Μονάδες 15

ΘΕΜΑ 2

2.1

$$y = 4 + x^{**2} + 2*(x+1)/(x^{**3}+1) - \text{abs}(x)$$

2.2

- α. Αποδεκτό
- β. Μη Αποδεκτό
- γ. Μη Αποδεκτό
- δ. Αποδεκτό
- ε. Αποδεκτό
- στ. Μη αποδεκτό

2.3

```
1  m= input(' Δώσε το βάρος σου (σε κιλά): ')
2  h= input(' Δώσε το ύψος σου (σε μέτρα): ')
3  dms=m/h**2
4  if dms<18.5:
5      print ' Ελλιποβαρής '
6  elif dms >= 18.5 and dms <= 24.99:
7      print ' Φυσιολογικό Βάρος '
8  elif dms >= 25.00 and dms <= 29.99:
9      print ' Προ-παχυσαρκία '
10 elif dms >= 30.00 and dms <= 34.99:
11     print ' Παχυσαρκία τύπου I '
12 elif dms >= 35.00 and dms <= 39.99:
13     print ' Παχυσαρκία τύπου II '
14 else:
15     print ' Παχυσαρκία τύπου III '
```

ΘΕΜΑ 4

Κατά την Παγκόσμια ημέρα της Αποταμίευσης, ένας καθηγητής Πληροφορικής, θέλοντας να κεντρίσει το ενδιαφέρον των μαθητών του, τους έβαλε το ακόλουθο κουίζ:

Σε πόσες ημέρες θα καταφέρουν να αποταμιεύσουν το ποσό των 100€ αν ξεκινήσουν την 1^η ημέρα από το αρχικό ποσό του ενός λεπτού του ευρώ (0.01 €) και κάθε επόμενη μέρα προστίθεται (αποταμιεύεται) στον κουμπαρά ποσό διπλάσιο από την προηγούμενη (την 1^η ημέρα 0.01€, τη 2^η ημέρα 0.02€, την 3^η μέρα 0.04€, κ.ο.κ). Για να προσομοιώσουν την διαδικασία της αποταμίευσης, ζήτησε από τους μαθητές να γράψουν ένα πρόγραμμα σε Python το οποίο:

4.1. Για κάθε μέρα να εμφανίζει το ποσό που έχει ήδη αποταμιευθεί στο κουμπαρά (π.χ. για την 1^η ημέρα θα εμφανίζει το μήνυμα «Στο κουμπαρά έχει αποταμιευθεί το ποσό των 0.01€»), μέχρι να συγκεντρωθεί το συνολικό ποσό των 100€.

Μονάδες 10

4.2. Να εμφανίζει πόσες μέρες χρειάστηκαν για ν' αποταμιευθεί το ποσό των 100€.

Μονάδες 5

4.3. Να υπολογίζει κι εμφανίζει κατά πόσο ξεπεράστηκε το ποσό των 100€.

Μονάδες 5

4.4 Τέλος, το πρόγραμμα θα πρέπει να υπολογίζει κι εμφανίζει ποιο ήταν το ποσό που τοποθετήθηκε στον κουμπαρά την τελευταία μέρα.

Μονάδες 5

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
    print b
```

ΘΕΜΑ 4

```
1      apot=0      #Αρχικό ποσό που έχει τοποθετηθεί στο κουμπαρά
2      poso=0.01   #Ποσό 1ης ημέρας αποταμίευσης
3      meres=0     # Ημέρες για να συγκεντρωθεί το ποσό των 100€
4      while apot<100: # Όσο το ποσό που έχει αποταμιευθεί είναι κάτω από 100€
5          meres=meres+1      #συνολικές ημέρες
6          apot=apot+poso     #το ποσό που έχει αποταμιευθεί στο κουμπαρά
7          print "Μέσα στο κουμπαρά υπάρχει το ποσό των ",apot, "€" #4.1
8          poso=poso*2        #Το ποσό που πρόκειται να αποταμιευθεί την επόμενη ημέρα
9
10     print "Το ποσό των ",apot," € συγκεντρώθηκε σε", meres," ημέρες" #4.2
11     print "Τελικά ξεπεράστηκε το ποσό των 100€ κατά", apot-100, " €" #4.3
12     print "Το τελευταίο ποσό που μπήκε στο κουμπαρά είναι",apot-(poso/2) #4.4
```

ΘΕΜΑ 2

2.1. Να συμπληρώσετε τα κενά στο παρακάτω πρόγραμμα, ώστε να διαβάξει δύο αριθμούς, να καλεί συνάρτηση η οποία δέχεται τους δύο αριθμούς κι επιστρέφει το άθροισμα και την διαφορά τους.

```
1      def praxeis ( (1) ):
2          s = x+y
3          d = abs(x-y)
4          (2)
5
6      #main program
7      (3) =input('Δώσε δύο αριθμούς')
8      (4) = (5) (a,b)
9      print ath,diaf
```

Να γράψετε στο γραπτό σας τους αριθμούς (1), (2), (3), (4) και (5) που αντιστοιχούν στα κενά του παραπάνω προγράμματος και δίπλα σε κάθε αριθμό, αυτό που πρέπει να συμπληρωθεί, ώστε το τμήμα προγράμματος να εκτελεί σωστά την λειτουργία του.

Μονάδες 5

2.2. Δίνεται το παρακάτω πρόγραμμα σε Python το οποίο διαβάξει ένα αριθμό και εμφανίζει αν ο αριθμός είναι θετικός, αρνητικός ή μηδενικός.

```
x=int(input('Δώσε αριθμό'))
```

```
    if x>=0:
        print "Ο αριθμός»,x,"είναι θετικός"
    elif x<=0:
        print "Ο αριθμός»,x,"είναι αρνητικός"
    else:
        print "Ο αριθμός»,x,"είναι μηδενικός"
```

Το πρόγραμμα έχει κάποιο λογικό λάθος. Να το περιγράψετε υποδεικνύοντας μια τιμή εισόδου για την οποία το πρόγραμμα δεν εμφανίζει το σωστό αποτέλεσμα. Να ξαναγράψετε στο γραπτό σας το πρόγραμμα διορθωμένο και με αποκλειστική χρήση της εντολής if (Απλή ΑΝ).

Μονάδες 10

2.3. Δίνεται το παρακάτω πρόγραμμα σε Python:

```
1      x=3
2      y=15
3      s=0
4      p=0
5      while y>0:
6          p=p+1
7          if y%2==1:
8              s=s+x
9              y=y/2
10             x=x*2
11             print x,y
12     print s,p
```

Επίσης δίνεται το παρακάτω υπόδειγμα πίνακα τιμών:

	x	y	s	p	Έξοδος
Αρχικές Τιμές	3	15	0	0	
1η επανάληψη	...	...	...	...	...
...	...	...	...	...	...

Να μεταφέρετε στο γραπτό σας τον παραπάνω πίνακα και να συμπληρώσετε τις τιμές των x , y , s , p και τι εμφανίζεται στην οθόνη σαν αποτέλεσμα της εκτέλεσης των εντολών του προγράμματος, προσθέτοντας μια νέα γραμμή στον πίνακα για κάθε επόμενη επανάληψη.

Μονάδες 10

ΘΕΜΑ 2

2.1

- (1). x, y
- (2). return s, d
- (3). a, b
- (4). ath, diaf
- (5). praxeis

2.2

Το πρόβλημα εντοπίζεται όταν δώσω την τιμή 0. Όταν στη μεταβλητή x δώσουμε την τιμή 0, ενώ το πρόγραμμα θα πρέπει να εμφανίσει το μήνυμα “Ο αριθμός 0 είναι μηδενικός”, εμφανίζει “Ο αριθμός 0 είναι θετικός”. Αυτό συμβαίνει διότι η πρώτη συνθήκη της εντολής if ελέγχει αν ο αριθμός x είναι μεγαλύτερος ή ίσος του 0. Για την τιμή 0 η συνθήκη αυτή είναι True και εκτελείται η εντολή που την ακολουθεί (**print** “Ο αριθμός”,x,”είναι θετικός”). Στη συνέχεια και λόγω του τρόπου που λειτουργεί η πολλαπλή δομή επιλογής, δεν εξετάζονται οι επόμενες συνθήκες και το πρόγραμμα τερματίζεται.

Μετά την διόρθωση και τη μετατροπή της **πολλαπλής if σε απλές if (AN)**, το πρόγραμμα γίνεται:

```
x=int(input('Δώσε αριθμό'))
if x>0:
    print “Ο αριθμός”,x,”είναι θετικός”
if x<0:
    print “Ο αριθμός”,x,”είναι αρνητικός”
if x==0:
    print “Ο αριθμός”,x,”είναι μηδενικός”
```

2.3

	x	y	s	p	Έξοδος
Αρχικές Τιμές	3	15	0	0	
1η επανάληψη	6	7	3	1	6, 7
2η επανάληψη	12	3	9	2	12, 3
3η επανάληψη	24	1	21	3	24, 1
4η επανάληψη	48	0	45	4	48, 0
					45, 4

ΘΕΜΑ 4

Στην Α΄ Φάση των Σχολικών Αγώνων της Περιφέρειας Πελοποννήσου, στο αγώνισμα της Σφύρας συμμετείχαν 95 μαθητές/τριες από όλα τα σχολεία Δευτεροβάθμιας Εκπαίδευσης που ανήκουν στην Περιφέρεια αυτή. Στη Β΄ Φάση θα προκριθούν οι μαθητές/τριες που η επίδοσή τους ξεπέρασε το όριο των 50 μέτρων, για τα ΑΓΟΡΙΑ, και το όριο των 40 μέτρων, για τα ΚΟΡΙΤΣΙΑ. Να γράψετε πρόγραμμα σε Python το οποίο:

4.1. Για κάθε μαθητή/τρια που αγωνίσθηκε, θα διαβάζει το ονοματεπώνυμο του/της, το φύλο του/της (Α΄ για αγόρι ή Κ΄ για κορίτσι), την καλύτερη του/της επίδοση (σε μέτρα) και το όνομα του σχολείου στο οποίο φοιτά. Η εισαγωγή των δεδομένων ολοκληρώνεται όταν δοθεί ως ονοματεπώνυμο η τιμή “ΤΕΛΟΣ ΑΓΩΝΑ”. Θεωρήστε ότι όλες οι τιμές εισόδου είναι ορθές και δεν απαιτείται έλεγχος εγκυρότητας τιμών.

Μονάδες 4

4.2. Τα στοιχεία όσων μαθητών/τριών πέρασαν στην επόμενη Β΄ Φάση θα καταχωρίζονται στις λίστες PROK_A (για τα αγόρια) και PROK_K (για τα κορίτσια), αντίστοιχα, ως εξής: Για το 1ο αγόρι που ξεπέρασε το όριο, στη θέση PROK_A[0] θα αποθηκεύεται το ονοματεπώνυμό του, στη θέση PROK_A[1] η καλύτερη επίδοσή του και στη θέση PROK_A[3] το όνομα του σχολείου του, και με τον ίδιο τρόπο θα συνεχιστεί η καταχώρηση για κάθε αγόρι που προκρίθηκε στη Β΄ Φάση. Με την ίδια διαδικασία θα καταχωρίζονται στη λίστα PROK_K τα στοιχεία για τα κορίτσια που προκρίθηκαν στην Β΄ Φάση. Θεωρήστε ότι προκρίθηκαν στη Φάση Β΄ τουλάχιστον ένα αγόρι και ένα κορίτσι.

Μονάδες 6

4.3. Θα υπολογίζει την καλύτερη επίδοση που καταμετρήθηκε στα κορίτσια και θα εμφανίζει το όνομα της μαθήτριας, την επίδοσή της και το όνομα του σχολείου της, με κατάλληλα διαμορφωμένο μήνυμα.

Μονάδες 5

4.4. Θα εμφανίζει το μήνυμα “Στη Β΄ Φάση προκρίθηκαν οι μαθητές:” και το ονοματεπώνυμο, την επίδοση, το όνομα του σχολείου του κάθε μαθητή σε διαφορετική γραμμή. Όμοια θα εμφανίζει “Στη Β΄ Φάση προκρίθηκαν οι μαθήτριες:” με τα ατομικά στοιχεία κάθε μαθήτριας σε διαφορετική γραμμή.

Μονάδες 10

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1  kal=-1    #4.3 Αρχικοποίηση μέγιστου
2  PROK_A=[] # 4.2 Αρχικοποίηση λίστα προκριθέντων μαθητών
3  PROK_K=[] #4.2 Αρχικοποίηση λίστα προκριθέντων μαθητριών
4  #4.1 Εισαγωγή ονοματεπώνυμου μαθητή/τριας
5  full_name=raw_input('Δώσε το ονοματεπώνυμο του/της μαθητή/τριας ')
6  while full_name!="ΤΕΛΟΣ ΑΓΩΝΑ": #4.1 διάβασε δεδομένα μέχρι να δοθεί "ΤΕΛΟΣ ΑΓΩΝΑ"
7      fylo=raw_input('Δώσε το φύλο '+' A '+'ή '+' K') #4.1 Εισαγωγή φύλου
8      epidosi=float(input('Δώσε την επίδοση του/της '+full_name)) #4.1 Εισαγωγή επίδοσης
9      name_sch=raw_input('Δώσε το όνομα του σχολείου του/της '+full_name) #4.1 όνομα Σχολείου
10     if fylo=='A': #4.2.
11         if epidosi>50: #4.2 Έλεγχος αν το ΑΓΟΡΙ πέρασε το όριο των 50 μέτρων
12             PROK_A.append(full_name) #4.2 καταχωρήστε πρώτα το ονοματεπώνυμο στη λίστα
13             PROK_A.append(epidosi) #4.2 μετά την επίδοση
14             PROK_A.append(name_sch) #4.2 μετά το σχολείο
15         else:
16             if epidosi>40: #4.2 Έλεγχος αν το ΚΟΡΙΤΣΙ πέρασε το όριο των 40 μέτρων
17                 PROK_K.append(full_name) #4.2 καταχωρήστε πρώτα το ονοματεπώνυμο στη λίστα
18                 PROK_K.append(epidosi) #4.2 μετά την επίδοση
19                 PROK_K.append(name_sch) #4.2 μετά το σχολείο
20
21         if epidosi>kal: # 4.3 εύρεση καλύτερης επίδοσης ΚΟΡΙΤΣΙΟΥ
22             kal=epidosi # 4.3
23             meg_name=full_name #4.3 ονοματεπώνυμο μαθήτριας με την καλύτερη επίδοση
24             meg_sch=name_sch #4.4 όνομα σχολείου μαθήτριας
25             full_name=raw_input('Δώσε το ονοματεπώνυμο του επόμενου/νης μαθητή/τριας ')
26     print 'Η καλύτερη επίδοση στα ΚΟΡΙΤΣΙΑ είναι: ',kal,' και την έχει η ',meg_name #4.3
27     print ' του σχολείου ', meg_sch
28     print '-----'
29     print 'Στην επόμενη Β' Φάση προκρίθηκαν οι μαθητές:' #4.4. ποιοί πέρασαν στη επόμενη φάση
30     for i in range(0, len(PROK_A), 3):
31         print PROK_A[i], ' με επίδοση ',PROK_A[i+1], ' του σχολείου ',PROK_A[i+2]
32     print '-----'
33     print 'Στην επόμενη Β' Φάση προκρίθηκαν οι μαθήτριες:' #4.4. ποιές πέρασαν στη επόμενη φάση
34     for i in range(0, len(PROK_K), 3):
35         print PROK_K[i], ' με επίδοση ', PROK_K[i+1], ' του σχολείου ',PROK_K[i+2]
```

ΘΕΜΑ 4

Μια ασφαλιστική εταιρεία, υπολογίζει τα ετήσια ασφάλιστρα που πρέπει να καταβάλει ο/η οδηγός ενός επιβατικού οχήματος λαμβάνοντας υπόψη τους παρακάτω παράγοντες:

- Τους φορολογητέους ίππους του οχήματος, σύμφωνα με τον πίνακα:

Ίπποι οχήματος	Ασφάλιστρα εξαμήνου
μέχρι 5	110 €
6 - 10	150 €
11 - 14	190 €
15 - 19	230 €
20 και πάνω	260 €

- Την ηλικία του οδηγού ως εξής, αν ο οδηγός είναι από 18 μέχρι 25 ή πάνω από 65 ετών, τότε τα ετήσια ασφάλιστρα που προκύπτουν από τον παραπάνω πίνακα αυξάνονται κατά 12%. Για οποιαδήποτε άλλη ηλικία οδηγού τα ετήσια ασφάλιστρα αυξάνονται 5.5%.
- Το είδος της ασφάλισης (μεικτή ή απλή). Η μεικτή ασφάλιση επιβαρύνει 10% τα ετήσια ασφάλιστρα του παραπάνω πίνακα.

Να αναπτύξετε πρόγραμμα στην γλώσσα προγραμματισμού Python το οποίο

4.1. Να διαβάζει την ηλικία του/της οδηγού, τους ίππους του οχήματος και το είδος της ασφάλισης το οποίο επιθυμεί να ασφαλίσει. Η ηλικία του/της οδηγού πρέπει να είναι 18 ετών και πάνω και το είδος ασφάλισης μεικτή ή απλή. Απαιτείται ο έλεγχος δεδομένων εισόδου για την ηλικία και το είδος ασφάλισης. Θεωρείστε ότι οι ίπποι είναι θετικός ακέραιος αριθμός.

Μονάδες 7

4.2. Να υπολογίζει και να εμφανίζει τα ετήσια καθαρά ασφάλιστρα του οχήματος σύμφωνα με τον παραπάνω πίνακα.

Μονάδες 7

4.3. Να υπολογίζει την ετήσια αύξηση των ασφαλιστρών λαμβάνοντας υπόψη αρχικά την ηλικία του οδηγού και κατόπιν το είδος της ασφάλισης.

Μονάδες 7

4.4. Να υπολογίζει και να εμφανίζει τα **συνολικά** ετήσια ασφάλιστρα μετά το μήνυμα «Τα συνολικά ετήσια ασφάλιστρα είναι».

Μονάδες 4

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1  #4.1 Διαβάζει την ηλικία του/της οδηγού
2  age=int(input('Δώσε ηλικία του/της οδηγού: '))
3  # 4.1 Ελέγχει αν η ηλικία είναι 18 και άνω
4  while age<18:
5      age=int(input('Δώσε σωστά την ηλικία του/της οδηγού: '))
6  # 4.1 Διαβάζει τους φορολογητέους ίππους του οχήματος
7  f_ipoi=int(input('Δώσε φορολογητέους ίππους οχήματος: '))
8  #4.1 Διαβάζει το είδος της ασφάλισης
9  eidos=raw_input('Δώσε είδος ασφάλισης (μεικτή/απλή: ')
10 #4.1 Ελέγχει αν το είδος ασφάλισης είναι μεικτή ή απλή
11 while eidos!= 'μεικτή' and eidos!= 'απλή':
12     eidos=raw_input('Δώσε σωστό είδος ασφάλισης (μεικτή/απλή): ')
13 #4.2 Υπολογισμός των ετήσιων καθαρών ασφαλίσεων
14 if f_ipoi<=5:
15     clean_asf=110*2
16 elif f_ipoi<=10:
17     clean_asf=150*2
18 elif f_ipoi<=14:
19     clean_asf=190*2
20 elif f_ipoi<=19:
21     clean_asf=230*2
22 else:
23     clean_asf=260*2
24 #4.2 Εμφάνιση των ετήσιων καθαρών ασφαλίσεων
25 print 'Τα καθαρά ετήσια ασφαλίσει του οχήματος είναι: ',clean_asf
26 #4.3 Υπολογισμός των αυξήσεων στα ασφαλίσει λόγω ηλικίας
27 if age<=25 or age>=65:
28     asf_age=clean_asf*12/100
29 else:
30     asf_age=clean_asf*5.5/100
31 #4.3 Υπολογισμός των αυξήσεων στα ασφαλίσει λόγω είδους ασφάλισης
32 if eidos=='μεικτή':
33     asf_eidos=clean_asf*10/100
34 else:
35     asf_eidos=0
36 t_incr=asf_age+asf_eidos #4.3 υπολογισμός της συνολικής ετήσιας αύξησης
37 #4.4 Υπολογισμός και εμφάνιση των ετήσιων συνολικών ασφαλίσεων
38 asf_total=clean_asf+asf_age+asf_eidos
39 print 'Τα ετήσια συνολικά ασφαλίσει είναι: ',asf_total
```

ΘΕΜΑ 2

2.1. Ο τομέας της Επικοινωνίας Ανθρώπου-Υπολογιστή(Ε.Α.Υ.) ασχολείται με το σχεδιασμό, την υλοποίηση, και την αξιολόγηση διαδραστικών υπολογιστικών συστημάτων προορισμένων για ανθρώπινη χρήση και τη μελέτη σημαντικών φαινομένων γύρω από αυτά. Ο βασικός στόχος της Ε.Α.Υ. είναι η αυξημένη ευχρηστία του συστήματος. Επιλέξτε από τα παρακάτω όλα όσα θεωρείτε ότι πρέπει να χαρακτηρίζουν ένα περιβάλλον, μέσω του οποίου επιτυγχάνεται η επικοινωνία ενός χρήστη με ένα σύστημα υπολογιστή.

1. ευκολία εκμάθησης
2. ευκολία ανάδρασης του έργου
3. υψηλή απόδοση εκτέλεσης έργου
4. ανίχνευση λογικών λαθών
5. διερεύνηση ποιότητας
6. χαμηλή συχνότητα εμφάνισης λαθών χρήστη
7. ευκολία συγκράτησης της γνώσης της χρήσης του
8. υποκειμενική ικανοποίηση του χρήστη
9. αντικειμενική ευχρηστίας του συστήματος

Μονάδες 5

2.2. Στον παρακάτω πίνακα η στήλη Α περιέχει αριθμητικές πράξεις (παραστάσεις) ενώ η στήλη Β περιέχει την αποτίμησή τους (αποτελέσματα τους) στο IDLE της Python. Να γράψετε στο τετράδιό σας τους αριθμούς 1,2,3,4,5 της **Στήλης Α** και δίπλα το γράμμα α, β, γ, δ, ε της **Στήλης Β**, που δίνει τη σωστή αντιστοιχία.

Στήλη Α	Στήλη Β
1. $2\%3$	α. -2
2. $11\%3-2*2$	β. 5
3. $15 * 2 / 4$	γ. 2
4. $2*(5\%3)+4/(1+3)$	δ. 6
5. $2*5/3*2$	ε. 7

Μονάδες 10

2.3. Δίνεται το παρακάτω τμήμα κώδικα σε γλώσσα προγραμματισμού Python. Τι τιμές θα εμφανιστούν κατά την εκτέλεση του αν δοθούν οι τιμές:

α) 3, β) 3.5, γ) 8, δ) 4.8 ε) 2.9

```
num=int(input("Δώσε αριθμό"))
```

```
check=num%2
```

```
if check==0:
```

```
    print "ο αριθμός είναι άρτιος"
```

```
else:
```

```
    print "ο αριθμός είναι περιττός"
```

Να γράψετε στο τετράδιο σας τα α, β, γ, δ, ε και δίπλα την τιμή που θα εμφανιστεί σύμφωνα με τις παραπάνω τιμές εισόδου.

Μονάδες 10

ΘΕΜΑ 2

2.1. Σωστά είναι τα 1, 3, 6, 7, 8

2.2.

$$1 \rightarrow \gamma$$

$$2 \rightarrow \alpha$$

$$3 \rightarrow \varepsilon$$

$$4 \rightarrow \beta$$

$$5 \rightarrow \delta$$

2.3.

α) Ο αριθμός είναι περιττός

β) Ο αριθμός είναι περιττός

γ) Ο αριθμός είναι άρτιος

δ) Ο αριθμός είναι άρτιος

ε) Ο αριθμός είναι άρτιος

ΘΕΜΑ 2

2.1. Ποιο θα είναι το αποτέλεσμα της εκτέλεσης του παρακάτω τμήματος προγράμματος σε Python;

```
1    a=[6, 3, 4, 55, 0]
2    b=a[:]
3    print b
```

Μονάδες 5

2.2. Δίνονται οι συμβολοσειρές της Python **a="Η Πληροφορική είναι "** και **b="προγραμματισμός "** και το παρακάτω τμήμα προγράμματος σε Python. Να γράψετε στο τετράδιό σας το αποτέλεσμα που εμφανίζεται σε κάθε εντολή εκτύπωσης (εντολές 3-7).

```
1    c=a+b
2    d=b*2
3    print c
4    print d
5    print len(a)
6    print a[2]
7    print b[0:9]
```

Μονάδες 10

2.3. Δίνεται ο παρακάτω κώδικας σε γλώσσα προγραμματισμού Python και οι λίστες a, b. Τι τιμές θα εμφανιστούν κατά την εκτέλεση του κώδικα;

```
1    a=[3,4,9,67,5]
2    b=a+[]
3    a[0]=a[1]=a[2]=a[3]=a[4]=1
4    print a
5    print b
6    print a+b
7    print b[-2]
8    print a[0:2]+b[2:4]
```

Μονάδες 10

ΘΕΜΑ 2

2.1.

[6, 3, 4, 55, 0]

2.2.

Εντολή 3: Η Πληροφορική είναι προγραμματισμός

Εντολή 4: προγραμματισμός προγραμματισμός

Εντολή 5: 20

Εντολή 6: Π

Εντολή 7: προγραμμα

2.3.

Εντολή 4: [1, 1, 1, 1, 1]

Εντολή 5: [3, 4, 9, 67, 5]

Εντολή 6: [1, 1, 1, 1, 1, 3, 4, 9, 67, 5]

Εντολή 7: 67

Εντολή 8: [1, 1, 9, 67]

ΘΕΜΑ 4

Τα ετήσια τέλη κυκλοφορίας ενός αυτοκινήτου υπολογίζονται με βάση τον κυβισμό και το είδος του αυτοκινήτου (συμβατικό ή υβριδικό), σύμφωνα με τον παρακάτω πίνακα:

Κυβισμός (σε κ.εκ)	Συμβατικά	Υβριδικά
0 - 1100	230	100
1101 - 1400	290	120
1400 - 1800	350	150
1801 και πάνω	480	200

Να αναπτύξετε πρόγραμμα στην γλώσσα προγραμματισμού Python το οποίο για 100 αυτοκίνητα:

4.1. Να διαβάσει τον αριθμό κυκλοφορίας και τον κυβισμό τους, ελέγχοντας ότι ο κυβισμός είναι θετικός αριθμός. Στην περίπτωση που εισάγεται λανθασμένη τιμή για τον κυβισμό, το πρόγραμμα να ζητάει ξανά τιμή μέχρι να δοθεί έγκυρη.

Μονάδες 4

4.2. Να διαβάσει το είδος του κάθε αυτοκινήτου ελέγχοντας αν είναι συμβατικό ή υβριδικό (Σ/Υ). Στην περίπτωση που εισάγεται λανθασμένη τιμή το πρόγραμμα να ζητάει ξανά τιμή μέχρι να δοθεί έγκυρη.

Μονάδες 4

4.3. Να υπολογίζει με χρήση της συνάρτησης `teli_kykl`, που θα κατασκευαστεί, τα ετήσια τέλη κυκλοφορίας και να εμφανίζει στο κυρίως πρόγραμμα κατάλληλο μήνυμα, με τον αριθμό κυκλοφορίας κάθε αυτοκινήτου και το ποσό των ετησίων τελών κυκλοφορίας του.

Μονάδες 13

4.4. Να υπολογίζει και να εμφανίζει τα πλήθη των συμβατικών και των υβριδικών αυτοκινήτων.

Μονάδες 4

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1  #4.3 συνάρτηση υπολογισμού τελών κυκλοφορίας
2  def teli_kykl( kyb , eidos):
3      if eidos=='Σ':#έλεγχος για ειδος αυτοκινήτου
4          if kyb<=1100: #έλεγχος για αριθμό κυβικών
5              poso=230
6          elif kyb<=1400:#έλεγχος για αριθμό κυβικών
7              poso=290
8          elif kyb<=1800:#έλεγχος για αριθμό κυβικών
9              poso=350
10         else:      #έλεγχος για αριθμό κυβικών
11             poso=480
12     else:
13         if kyb<=1100: #έλεγχος για αριθμό κυβικών
14             poso=100
15         elif kyb<=1400:#έλεγχος για αριθμό κυβικών
16             poso=120
17         elif kyb<=1800:#έλεγχος για αριθμό κυβικών
18             poso=150
19         else:      #έλεγχος για αριθμό κυβικών
20             poso=200
21     return poso #επιστροφή ποσού τελών κυκλοφορίας
22
23 #κυρίως πρόγραμμα
24 metr=0 #4.4 αρχικοποίηση μετρητή συμβατικών αυτοκινήτων
25 #επαναληπτική διαδικασία για τα 100 αυτοκίνητα
26 for i in range (100):
27     ar=raw_input('Δώσε αριθμό κυκλοφορίας ')#4.1 Διαβάζει αριθμό κυκλοφορίας
28     kyb=int(input('Δώσε κυβισμό ')) #4.1 διαβάζει κυβισμό αυτοκινήτου
29     while kyb<=0: #4.1 ελέγχει μήπως ο κυβισμός είναι αρνητικός αριθμός
30         kyb=int(input('Δώσε σωστό κυβισμό '))
31     #4.2 διάβασμα και έλεγχος για το είδος του αυτοκινήτου
32     eidos=raw_input ('Δώσε ειδος αυτοκινήτου Σ/Υ(συμβατικό/υβριδικό) ')
33     while eidos !='Σ' and eidos !='Υ':
34         eidos=raw_input ('Δώσε σωστά το ειδος αυτοκινήτου Σ/Υ(συμβατικό/υβριδικό) ')
35     # 4.3 κλήση της συνάρτησης για τα τέλη κυκλοφορίας
36     tel_poso = teli_kykl(kyb , eidos)
37     print 'τα τέλη κυκλοφορίας του αυτοκινήτου ', ar , ' είναι ',tel_poso,' ευρώ'
38     # 4.4 έλεγχος για το είδος. Αν είναι συμβατικό, αύξηση μετρητή
39     if eidos=='Σ':
40         metr=metr+1
41     #4.4 εμφάνιση πλήθους συμβατικών και υβριδικών αυτοκινήτων
42     print 'Το πλήθος των συμβατικών αυτοκινήτων είναι ', metr, ' και των υβριδικών ',100-metr
```

ΘΕΜΑ 2

2.1. Δίνεται το παρακάτω τμήμα προγράμματος στη γλώσσα προγραμματισμού Python. Τι τιμές θα εμφανιστούν κατά την εκτέλεση του;

```
1   a='Καλή'
2   b='Άνοιξη'
3   msg=a+' '+b
4   print 2*a
5   print msg
```

Μονάδες 6

2.2. Δίνονται τα παρακάτω τμήματα προγράμματος Α, Β, Γ στη γλώσσα προγραμματισμού Python. Σε ποιο από αυτά τα τμήματα προγράμματος εμφανίζεται το μήνυμα «δεν μπορείς να ψηφίσεις» αν το α είναι μικρότερο του 18, «μπορείς να μην ψηφίσεις» αν το α είναι μεγαλύτερο του 80, ενώ όταν είναι ενδιάμεσα εμφανίζει «ψηφίζεις».

```
A
1   a=int(input('Δώσε ηλικία: '))
2   if a>18 :
3       print 'ψηφίζεις'
4   elif a>80 :
5       print 'μπορείς να μην ψηφίσεις'
6   else :
7       print 'δεν μπορείς να ψηφίσεις'
```

```
B
1   a=int(input('Δώσε ηλικία: '))
2   if a<18 :
3       print 'δεν μπορείς να ψηφίσεις'
4   elif a<80 :
5       print 'ψηφίζεις'
6   else :
7       print 'μπορείς να μην ψηφίσεις'
```

```
Γ
1   a=int(input('Δώσε ηλικία: '))
2   if a>80 :
3       print 'μπορείς να μην ψηφίσεις'
4   elif a<18 :
5       print 'δεν μπορείς να ψηφίσεις'
6   else :
7       print 'ψηφίζεις'
```

Μονάδες 9

2.3. Να συμπληρωθούν με τον κατάλληλο κώδικα τα αριθμημένα κενά (1-8) στο παρακάτω πρόγραμμα στη γλώσσα προγραμματισμού Python, ώστε το πρόγραμμα να διαβάζει δύο λίστες A και B, 10 θέσεων η καθεμία, και να δημιουργεί και εκτυπώνει μια τρίτη λίστα C, επίσης 10 θέσεων, της οποίας κάθε στοιχείο αποτελεί άθροισμα των αντίστοιχων στοιχείων των λιστών A και B.

```
1   A = []
2   B = []
3   C = []
4   for i in range(10):
5       x = int(input("Δώσε ακέραιο για τη λίστα A: "))
6       A.append(-1-)
7   for i in range(-2-):
8       y = -3-
9       -4-
10  -5-range(-6-):
11  -7-
12  print(-8-)
```

Μονάδες 10

ΘΕΜΑ 2

2.1.

Οι τιμές που θα εκτυπωθούν είναι:

ΚαλήΚαλή

Καλή Άνοιξη

2.2.

Τις ζητούμενες τιμές θα εκτυπώσει το τμήμα Γ.

2.3. Το πρόγραμμα, με τα 8 κενά συμπληρωμένα, είναι το ακόλουθο:

```
1   A = []
2   B = []
3   C = []
4   for i in range(10):
5       x = int(input("Δώσε ακέραιο για τη λίστα A: "))
6       A.append(x)
7   for i in range(10):
8       y = int(input("Δώσε ακέραιο για τη λίστα B: "))
9       B.append(y)
10  for i in range(10):
11      C.append(A[i] + B[i])
12  print(C)
```

ΘΕΜΑ 4

Για να συμμετάσχει κάποιος αθλητής ή αθλήτρια στον υπερμαραθώνιο των 245,3 χλμ «Σπάρταθλον» είναι απαραίτητο ή να έχει καλύψει την παραπάνω διαδρομή σε 36 ώρες σε προηγούμενη διοργάνωση «Σπάρταθλον» ή να έχει καλύψει απόσταση τουλάχιστον 100 χλμ. εντός 12 ωρών σε κάποια άλλη επίσημη διοργάνωση. Στην τρέχουσα διοργάνωση έχουν δηλώσει συμμετοχή 1800 αθλητές/τριες από όλο τον κόσμο. Να αναπτύξετε πρόγραμμα στην γλώσσα προγραμματισμού Python το οποίο:

4.1 Να διαβάζει για **κάθε** αθλητή/τρια το ονοματεπώνυμο, το φύλο ('Μ' για αθλητή, 'W' για αθλήτρια), την ηλικία, τη χώρα προέλευσης (π.χ. gr για Ελλάδα, fr για Γαλλία κ.ο.κ), αν έχει τερματίσει σε προηγούμενο «Σπάρταθλον» σε 36 ώρες ('Y'/'N') και τα περισσότερα χιλιόμετρα που έχει καλύψει σε προηγούμενη επίσημη διοργάνωση σε 12 ώρες. Μόνο για το φύλο απαιτείται έλεγχος των δεδομένων που εισάγονται.

Μονάδες 7

4.2 Να ελέγχει αν ο/η αθλητής/τρια έχει δικαίωμα συμμετοχής στον αγώνα «Σπάρταθλον» και να εμφανίζει, μετά το ονοματεπώνυμό του/της, το μήνυμα «Έχει δικαίωμα συμμετοχής» ή «Δεν έχει δικαίωμα συμμετοχής». Θεωρείστε ότι υπάρχει τουλάχιστον ένας αθλητής και μια αθλήτρια με δικαίωμα συμμετοχής στον αγώνα.

Μονάδες 5

4.3 Να υπολογίζει και να εμφανίζει το ποσοστό των Ελλήνων αθλητών και το ποσοστό των Ελληνίδων αθλητριών, οι οποίοι/ες έχουν δικαίωμα συμμετοχής στον αγώνα, σε σχέση με το σύνολο των αθλητών/τριών, οι οποίοι/ες, επίσης, έχουν δικαίωμα συμμετοχής στον αγώνα. Υποθέστε ότι υπάρχει τουλάχιστον ένας Έλληνας αθλητής και μια Ελληνίδα αθλήτρια με δικαίωμα συμμετοχής στον αγώνα.

Μονάδες 7

4.4 Να υπολογίζει και να εμφανίζει ξεχωριστά το μέσο όρο ηλικίας των αθλητών και των αθλητριών, οι οποίοι/ες έχουν δικαίωμα συμμετοχής στον αγώνα.

Μονάδες 6

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
    print b
```

ΘΕΜΑ 4

```
1  #Μετρητές για αθλητές/τριες που έχουν δικαίωμα συμμετοχής.
2  #Αρχικοποίηση
3  tot_m=0
4  tot_w=0
5  #Μετρητές για αθλητές/τριες απο Ελλάδα που έχουν δικαίωμα συμμετοχής.
6  #Αρχικοποίηση
7  gr_m=0
8  gr_w=0
9  #Αθροιστές για ηλικίες αθλητών και αθλητριών που έχουν δικαίωμα
10  συμμετοχής.
11  #Αρχικοποίηση
12  S_m=0
13  S_w=0
14  #4.1 επαναληπτική διαδικασία
15  for i in range (1800):
16      # 4.1 Διάβασμα ονοματεπώνυμου
17      onom=raw_input('Δώσε ονοματεπώνυμο αθλητή: ')
18      #4.1 Διάβασμα φύλου και έλεγχος έγκυρης τιμής
19      filo=raw_input('Δώσε με λατινικούς χαρακτήρες το φύλο (M/W) ')
20      while filo !='M' and filo!='W':
21          filo=raw_input('Δώσε σωστά με λατινικούς χαρακτήρες το φύλο (M/W): ')
22      #4.1 Διάβασμα ηλικίας
23      age=int(input(' Δώσε την ηλικία σε έτη του/της αθλητή/τριας: '))
24      #4.1 Διάβασμα εθνικότητας
25      nation=raw_input('Δώσε εθνικότητα, gr για Ελλάδα: ')
26      #4.1 Διάβασμα προηγούμενης επιτυχημένης συμμετοχής σε Σπάρταθλον
27      fin_before=raw_input(' Έχεις τερματίσει σε άλλο Σπάρταθλον; (Y/N): ')
28      #4.1 Διάβασμα χιλιομέτρων σε άλλη επίσημη διοργάνωση
29      max_km=int(input(' Δώσε αριθμό χιλιομέτρων σε 12 ώρες :'))
30      #4.2 Έλεγχος δικαιώματος συμμετοχής ή μη
31      if fin_before=='Y' or max_km>=100:
32          print 'Ο/η αθλητής/τρια ', onom, ' έχει δικαίωμα συμμετοχής'
```

```

32 #4.3 υπολογισμός συνολικών αθλητών και αθλητριών
33     if filo=='M':
34         tot_m=tot_m+1
35         S_m=S_m+ age # 4.4 για τον μέσο όρο της ηλικίας αθλητών
36 # 4.3 εύρεση Ελλήνων αθλητών που συμμετέχουν
37     if filo=='M' and nation == 'gr':
38         gr_m=gr_m+1
39     else:
40         tot_w=tot_w+1
41         S_w=S_w+age #4.4 για τον μέσο όρο της ηλικίας αθλητριών
42 # 4.3 εύρεση Ελληνίδων αθλητριών που συμμετέχουν
43     if filo=="W" and nation=='gr':
44         gr_w=gr_w+1
45     else:
46         print 'Ο/η αθλητής/τρια ', onom, ' δεν έχει δικαίωμα συμμετοχής'
47 #4.3 εμφάνιση ποσοστών ελληνικών συμμετοχών
48 print 'Ανάμεσα στους συμμετέχοντες: '
49 print 'Το ποσοστό των Ελλήνων αθλητών είναι: ', float(gr_m)/tot_m*100
50 print 'Το ποσοστό των Ελληνίδων αθλητριών είναι: ', float(gr_w)/tot_w*100
51 #4.4 εμφάνιση μέσου όρου ηλικίας αθλητών και αθλητριών
52 print 'Ο μέσος όρος ηλικίας των συμμετεχόντων αθλητών είναι :', float(S_m)/tot_m
53 print 'Ο ΜΟ ηλικίας όλων των συμμετεχουσών αθλητριών είναι :', float(S_w)/tot_w

```

ΘΕΜΑ 2

2.1. Να γράψετε στο γραπτό σας το παρακάτω τμήμα προγράμματος χρησιμοποιώντας της εντολή επανάληψης **for** αντί της εντολής επανάληψης **while** έτσι ώστε να εμφανίζει το ίδιο αποτέλεσμα.

```
1  i=0
2  while i<30:
3      i=i+3
4      print i
```

Μονάδες 13

2.2. Δίνεται το παρακάτω πρόγραμμα σε γλώσσα προγραμματισμού Python, το οποίο υπολογίζει και τυπώνει το άθροισμα των ακέραιων αριθμών (1+2+3+4+5+6+7+8+9+10)

```
1  # Πρόγραμμα άθροισμα ακεραίων αριθμών από το 1 έως το 10
2  a = (1)
3  for i in range((2) , (3)):
4      a = a+ (4)
```

Στο πρόγραμμα υπάρχουν υπογραμμισμένα κενά τα οποία έχουν αριθμηθεί. Να γράψετε στο γραπτό σας τους αριθμούς **1, 2, 3, και 4** που αντιστοιχούν στα κενά του παραπάνω προγράμματος και δίπλα σε κάθε αριθμό αυτό που πρέπει να συμπληρωθεί για να υλοποιείται σωστά το πρόγραμμα.

Μονάδες 8

2.3. Στη μεταβλητή A καταχωρίζεται η προφορική βαθμολογία ενός μαθήματος η οποία ελέγχεται να είναι στο διάστημα από 0 μέχρι 20 (Το 0 και το 20 είναι επιτρεπτές τιμές). Ποια από τις παρακάτω λογικές εκφράσεις ελέγχει αυτή τη συνθήκη;

- α. ($A > 0$) or ($A < 20$)
- β. ($A \geq 20$) and ($A \leq 20$)
- γ. ($A \geq 0$) and ($A \leq 20$)
- δ. ($A \geq 0$) or ($A \leq 20$)

Μονάδες 4

ΕΝΔΕΙΚΤΙΚΗ ΛΥΣΗ

ΘΕΜΑ 2

2.1.

```
1  for i in range (3,31,3):  
2      print i
```

2.2.

```
1  # Πρόγραμμα άθροισμα ακεραίων αριθμών από το 1 έως το 10  
2  a = 0  
3  for i in range(1 , 11):  
4      a = a+ i
```

2.3.

γ. ($B \geq 0$) and ($B \leq 20$)

ΘΕΜΑ 4

Στο νέο μισθολόγιο των δημοσίων υπαλλήλων χορηγείται μηνιαία οικογενειακή παροχή, ανάλογα με την οικογενειακή κατάσταση των υπαλλήλων, ως εξής:

ΟΙΚΟΓΕΝΕΙΑΚΗ ΚΑΤΑΣΤΑΣΗ ΥΠΑΛΛΗΛΟΥ	ΠΟΣΟ ΟΙΚΟΓΕΝΕΙΑΚΗΣ ΠΑΡΟΧΗΣ ΣΕ ΕΥΡΩ
Υπάλληλος με 1 παιδί	50
Υπάλληλος με 2 παιδιά	70
Υπάλληλος με 3 παιδιά	120
Υπάλληλος με 4 παιδιά	170
Υπάλληλος με >5 παιδιά	170 + 70€ για κάθε επιπλέον τέκνο

Να γράψετε πρόγραμμα σε γλώσσα Python το οποίο:

4.1. Να διαβάζει τον αριθμό των παιδιών ενός υπαλλήλου, ελέγχοντας την εγκυρότητα των δεδομένων εισόδου, δηλαδή ο αριθμός να είναι μεγαλύτερος ή ίσος με μηδέν.

Μονάδες 6

4.2. Θα καλεί συνάρτηση με όνομα `eridoma_teknwn` η οποία θα δέχεται τον αριθμό των παιδιών και θα υπολογίζει και θα επιστρέφει το ποσό επιδόματος που θα δικαιούται ο υπάλληλος σύμφωνα με τον παραπάνω πίνακα.

8 Μονάδες

4.3. Να εμφανίζει το ποσό του επιδόματος που αναλογεί στον υπάλληλο το οποίο θα εμφανίζεται μόνο στην περίπτωση που είναι διάφορο του 0.

Μονάδες 5

4.4. Θα εμφανίζει το μήνυμα ΤΡΙΤΕΚΝΗ ή ΠΟΛΥΤΕΚΝΗ οικογένεια στην αντίστοιχη περίπτωση.

Μονάδες 6

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
    print b
```

ΕΝΔΕΙΚΤΙΚΕΣ ΛΥΣΕΙΣ

ΘΕΜΑ 4

```
1  #Υπολογισμός οικογενειακού επιδόματος
2  def epidoma_tekwn(x):
3      if x==0:
4          epidoma=0
5      elif x==1:
6          epidoma=50 #για 1 παιδί
7      elif x==2:
8          epidoma=70 #για 2 παιδιά
9      elif x==3:
10         epidoma=120 #για 3 παιδιά
11     elif x==4:
12         epidoma=170 #για 4 παιδιά
13     else:
14         epidoma=170+(x-4)*70 #+70€ για κάθε επιπλέον τέκνο
15     return epidoma
16
17 #Κυρίως πρόγραμμα
18 ap=int(input('Δώσε αριθμό παιδιών '))
19 while ap<0:
20     print "Έδωσες λάθος αριθμό παιδιών"
21     ap=int(input('Δώσε αριθμό παιδιών '))
22     poso=epidoma_tekwn(ap)
23     if poso!=0:
24         print 'Δικαιούσαι οικογενειακό επίδομα ',poso, '€'
25     else:
26         print 'Δεν δικαιούσαι οικογενειακό επίδομα'
27     if ap==3:
28         print 'ΤΡΙΤΕΚΝΗ ΟΙΚΟΓΕΝΕΙΑ'
29     if ap>=4:
30         print 'ΠΟΛΥΤΕΚΝΗ ΟΙΚΟΓΕΝΕΙΑ'
```


ΘΕΜΑ 2

2.1. Δίνεται το παρακάτω τμήμα προγράμματος στη γλώσσα προγραμματισμού Python. Τι τιμές θα εμφανιστούν κατά την εκτέλεση του;

```
1    def pol(a,b):  
2        return a*b  
3    a='Όμορφη'  
4    b='πόλη'  
5    song=a+' '+b+' '  
6    print song  
7    print len(song)  
8    print pol(5,2)  
9    print pol(3,song)  
10   print (len(pol(2,song)))
```

Μονάδες 12

2.2. Δίνεται η παρακάτω λίστα στην γλώσσα προγραμματισμού Python **A=[5,10,15,20,25]**. Γράψτε στο τετράδιό σας ποια θα είναι η μορφή (περιεχόμενα) της λίστας A ή τι θα συμβεί, μετά την εφαρμογή **κάθε μίας** από τις παρακάτω εντολές:

Εντολή 1: >>>A[0]=0

Εντολή 2: >>>A[2]=100

Εντολή 3: >>>A[4]=200

Εντολή 4:>>>A[5]=400

Μονάδες 8

2.3. Δίνεται το παρακάτω τμήμα προγράμματος στη γλώσσα προγραμματισμού Python. Να γράψετε στο γραπτό σας το γράμμα α,β,γ,δ και ε και δίπλα το αποτέλεσμα που θα εμφανιστεί αν δοθούν σαν είσοδος οι τιμές:

α) ASPPP98,

β) ASP5642,

γ) AMJ4521,

δ) BBBB7,

ε) CCC0987

```
1 PINAKIDES=['ASP5642', 'AMJ4521', 'AAA4532', 'BBB5638', 'CCC0987']
2 numero=raw_input('Δώσε αριθμό κυκλοφορίας')
3 done=numero in PINAKIDES
4 print done
```

Μονάδες 5

ΘΕΜΑ 2

2.1. Οι τιμές που θα εμφανιστούν στις πέντε εντολές εμφάνισης θα είναι όπως παρακάτω, με την προϋπόθεση ότι εκτελείται σωστά ή εντολή 5.

Όμορφη πόλη

12

10

Όμορφη πόλη Όμορφη πόλη Όμορφη πόλη

24

2.2.

Εντολή 1: [0, 10, 15, 20, 25]

Εντολή 2: [0, 10, 100, 20, 25]

Εντολή 3: [0, 10, 100, 20, 200]

Εντολή 4: Μήνυμα λάθους, αφού το A[5] είναι έξω από τα όρια της λίστας.

2.3. Τα αποτελέσματα των δοθέντων τιμών εισόδου είναι όπως παρακάτω.

a) False,

b) True,

c) True,

d) False,

e) True

ΘΕΜΑ 4

Στην εθνική οδό Αθήνας Θεσσαλονίκης, σε σημείο με όριο ταχύτητας τα 80 χλμ/ώρα, ένα κλιμάκιο της τροχαίας ελέγχει την κίνηση των οχημάτων με ηλεκτρονικά μέσα, καταγράφοντας την ταχύτητα, το είδος των οχημάτων και τον αριθμό πινακίδας τους. Σε μια ημέρα καταγράφηκε η διέλευση 5500 οχημάτων.

Να αναπτύξετε πρόγραμμα στην γλώσσα προγραμματισμού Python το οποίο

4.1. Να διαβάζει για **κάθε** όχημα τον αριθμό κυκλοφορίας του, την ταχύτητα διέλευσης και το είδος του. Να ελέγχει για το είδος του εάν πρόκειται για αυτοκίνητο (**auto**), φορτηγό (**truck**) ή δίτροχο (**bike**). Σε περίπτωση λανθασμένης εισαγωγής να ζητά ξανά το είδος του τροχοφόρου. Να θεωρήσετε ότι γίνεται για κάθε είδος τροχοφόρου τουλάχιστον μια είσοδος δεδομένων.

Μονάδες 6

4.2. Για κάθε τροχοφόρο που υπερβαίνει το όριο ταχύτητας, να εμφανίζει τον αριθμό κυκλοφορίας του και την ταχύτητά του ως εξής: «Το όχημα με αρ. κυκλοφορίας», **A_K**, «είχε ταχύτητα», **TAXYTHTA**, «χλμ/ω».

Μονάδες 4

4.3. Να υπολογίζει και να εμφανίζει, **για κάθε είδος** τροχοφόρου, πόσα τροχοφόρα του είδους καταγράφηκαν με υπέρβαση του ορίου ταχύτητας.

Μονάδες 6

4.4. Να υπολογίζει και να εμφανίζει με κατάλληλο μήνυμα την μεγαλύτερη ταχύτητα που καταγράφηκε την συγκεκριμένη ημέρα **από κάθε είδος** τροχοφόρου ξεχωριστά.

Μονάδες 9

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1  # 4.3, 4.4 Αρχικοποίηση μετρητών είδους και μέγιστων
2  car=0
3  moto=0
4  fortigo=0
5  max_car=0
6  max_moto=0
7  max_fortigo=0
8  #4.1 Διαβάζει για κάθε όχημα τα ζητούμενα
9  for i in range(5500):
10     a_k=raw_input('Δώσε τον αριθμό κυκλοφορίας του οχήματος: ')
11     speed=int(input('Δώσε ταχύτητα οχήματος: '))
12     eidos=raw_input('Δώσε το είδος οχήματος (auto/truck/bike): ')
13  # 4.1 Έλεγχος του είδους οχήματος
14     while eidos!='auto' and eidos!='truck' and eidos!='bike':
15         eidos=raw_input('Δώσε σωστά το είδος οχήματος (auto/truck/bike): ')
16  #4.2 Έλεγχος υπέρβασης ταχύτητας
17     if speed>80:
18         print 'Το όχημα με αρ. κυκλοφορίας ', a_k, ' είχε ταχύτητα ', speed, 'χλμ/ω'
19  #4.3 Μετρητές τροχοφόρων με υπέρβαση ταχύτητας
20     if eidos=='auto':
21         car=car+1
22  #4.4 Εύρεση μέγιστης ταχύτητας αυτοκινήτων
23     if speed>max_car:
24         max_car=speed
25     elif eidos=='truck':
26         fortigo=fortigo+1
27  #4.4 Εύρεση μέγιστης ταχύτητας φορτηγών
28     if speed>max_fortigo:
29         max_fortigo=speed
30     else:
31         moto=moto+1
32  #4.4 Εύρεση μέγιστης ταχύτητας δίτροχων
33     if speed>max_moto:
34         max_moto=speed
35  #4.3 Εμφανίσεις υπερβάσεων
36     print 'Αυτοκίνητα που υπερέβησαν το όριο ταχύτητας: ',car
37     print 'Φορτηγά που υπερέβησαν το όριο ταχύτητας: ', fortigo
38     print 'Δίτροχα που υπερέβησαν το όριο ταχύτητας: ', moto
39  #4.4 Εμφανίσεις μεγαλύτερων ταχυτήτων ανά είδος τροχοφόρου
40     print 'Μεγαλύτερη ταχύτητα αυτοκινήτου: ', max_car
41     print 'Μεγαλύτερη ταχύτητα αυτοκινήτου: ', max_fortigo
42     print 'Μεγαλύτερη ταχύτητα αυτοκινήτου: ', max_moto
```

ΘΕΜΑ 2

2.1. Να γράψετε στο τετράδιο σας τι εμφανίζει το παρακάτω πρόγραμμα κατά την εκτέλεση του, όταν στην εντολή 13 πληκτρολογήσετε το 16.

```
1  x=27
2  x=x+1
3  print x
4  x=x*0.01
5  print x
6  c=x=z=5
7  print x,c,z
8  x,c=2,4
9  print x,c
10 a,b,t=1,2,"Μαίρη"
11 print a,b,t
12 print "Καλημέρα " + t
13 hl=input("Πόσων χρονών είσαι; ")
14 print "Η ", t, "είναι ", hl," ετών"
```

Μονάδες 14

2.2. Να γράψετε στο τετράδιό σας τον αριθμό της κάθε γραμμής όπως φαίνεται παρακάτω και δίπλα το αποτέλεσμα της πράξης, θεωρώντας ότι η μεταβλητή a=10 και η μεταβλητή b=20:

1. a==20
2. a==20 or a==10
3. a==10 and a==20
4. (b-a)>=20
5. a**2<=100 and (not a>=10)
6. a<=20 and b>10
7. (a!=10) or (b!=0)
8. a**2+b%2
9. a+ int(5.678)
10. a**2
11. pow(a,2)

Μονάδες 11

ΘΕΜΑ 2

2.1.

3. 28

5. 0.28

7. 5 5 5

8. 2 4

9. 1 2 Μαίρη

12. Καλημέρα Μαίρη

13. Πόσων χρονών είσαι; 16

14. Η Μαίρη είναι 16 ετών

2.2.

1. False

2. True

3. False

4. False

5. False

6. True

7. True

8. 100

9. 15

10. 100

11. 100

ΘΕΜΑ 4

Ένας καθηγητής μαθηματικών έχει 25 μαθητές, που τους έχει βάλει τεστ και τους βαθμολόγησε στην εκατοντάβαθμια κλίμακα.

Να αναπτύξετε πρόγραμμα στην γλώσσα Python το οποίο

4.1. Να διαβάζει το επίθετο κάθε μαθητή και το βαθμό του στο τεστ. Ο βαθμός είναι έγκυρος αν ο βαθμός βρίσκεται στο διάστημα $[0,100]$. Στη περίπτωση που ο βαθμός που εισάγεται είναι λανθασμένος το πρόγραμμα να ζητάει ξανά βαθμό μέχρι να δοθεί έγκυρος βαθμός.

Μονάδες 10

4.2. Να υπολογίζει και να εμφανίζει τον **μεγαλύτερο** βαθμό.

Μονάδες 5

4.3. Να υπολογίζει και να εμφανίζει τον **μικρότερο** βαθμό.

Μονάδες 5

4.4. Να υπολογίζει και να εμφανίζει τον **Μέσο Όρο** όλων των βαθμών.

Μονάδες 5

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```


ΘΕΜΑ 4

```
1 s=0    #αρχικοποίηση αθροιστή
2 max=0   #αρχικοποίηση μεγαλύτερου
3 min=100 #αρχικοποίηση μικρότερου
4 #εντολή επανάληψης για την εισαγωγή 25 βαθμών
5 for i in range(0,25):
6     onoma=raw_input("Δώσε επίθετο μαθητή: ")
7     b=input("Δώσε βαθμό μαθητή: ")
8     #έλεγχος για την εγκυρότητα του βαθμού
9     while b<0 or b>100:
10         print("Έδωσες λάθος βαθμό.")
11         b=input("Δώσε ξανά τον σωστό βαθμό: ")
12     s=s+b
13     #έλεγχος για την εύρεση του μεγαλύτερου
14     if b>max:
15         max=b
16     #έλεγχος για την εύρεση του μικρότερου
17     if b<min:
18         min=b
19 mo=s/25 #υπολογισμός του Μέσου Όρου
20 # Εκτύπωση του μεγαλύτερου
21 print "Ο Μεγαλύτερος βαθμός του τεστ είναι ο: ",max
22 # Εκτύπωση του μικρότερου
23 print "Ο Μικρότερος βαθμός του τεστ είναι ο: ",min
24 # Εκτύπωση του μέσου όρου
25 print "Ο Μέσος Όρος όλων των βαθμών είναι: ",mo
```


ΘΕΜΑ 2

2.1. Να γράψετε στο τετράδιο σας τον αριθμό των παρακάτω αριθμητικών πράξεων και δίπλα το αποτέλεσμα της αντίστοιχης πράξης.

1. $2*(3+1)**2$

2. $3**2/3$

3. $\text{int}((3+2)/2)**2/2$

4. $(3+2/2)**2/2$

5. $25/(2**2+1)$

6. $(200-100/2)/2$

7. $2*\text{int}(5/2)+5$

8. $2**4/2$

9. $2**(4/2)$

10. $\text{int}(7/3)**2$

Μονάδες 10

2.2.

A. Να γράψετε ισοδύναμο με το παρακάτω τμήμα προγράμματος Python, χρησιμοποιώντας τη δομή επανάληψης while.

```
1  x=int(input("Δώσε ένα αριθμό από το 1 μέχρι το 10:"))
2  for i in range(1,11):
3      gin=x*i
4      print x,"* ",i,"=",gin
```

Μονάδες 10

B. Ποιες τιμές θα εμφανιστούν κατά την διάρκεια εκτέλεσης του παραπάνω τμήματος προγράμματος, αν δώσουμε τον αριθμό 4;

Μονάδες 5

ΘΕΜΑ 2

2.1.

1. 32

2. 3

3. 2

4. 8

5. 5

6. 75

7. 9

8. 8

9. 4

10. 4

2.2.

A.

```
1      x=int(input("Δώσε ένα αριθμό από το 1 μέχρι το 10:"))
2      i=1      #Αρχικοποίηση του μετρητή i
3      while i<11:  #έλεγχος του i
4          # πολλαπλασιασμό του αριθμού x
5          gin=x*i
6          # εκτύπωση του γινομένου
7          print x,"* ",i, "=",gin
8          # αύξηση του μετρητή i κατά 1
9          i=i+1
```

B.

Εμφανίζει τη προπαίδεια από το 1 μέχρι και το 10 του αριθμού που δώσαμε από το πληκτρολόγιο. Οπότε για τον αριθμό 4 το πρόγραμμα εμφανίζει:

$$4 * 1 = 4$$

$$4 * 2 = 8$$

$$4 * 3 = 12$$

$$4 * 4 = 16$$

$$4 * 5 = 20$$

$$4 * 6 = 24$$

$$4 * 7 = 28$$

$$4 * 8 = 32$$

$$4 * 9 = 36$$

$$4 * 10 = 40$$

ΘΕΜΑ 4

Μια εταιρεία δημοσκοπήσεων θέτει σ' ένα δείγμα 200 πολιτών το ερώτημα «Χρησιμοποιείτε τους κάδους ανακύκλωσης; »

Να αναπτύξετε πρόγραμμα στην γλώσσα Python το οποίο:

4.1. Να διαβάζει για κάθε πολίτη το φύλο του (Α=Άνδρας, Γ=Γυναίκα) και να γίνεται έλεγχος εγκυρότητας των δεδομένων. Στη περίπτωση που το φύλο (Α ή Γ) που εισάγεται είναι λάθος το πρόγραμμα να ζητάει ξανά νέα τιμή μέχρι να δοθεί η σωστή.

Μονάδες 7

4.2. Να διαβάζει την απάντηση στο ερώτημα, η οποία μπορεί να είναι ΝΑΙ ή ΟΧΙ ή ΔΕΝ ΞΕΡΩ και να γίνεται έλεγχος εγκυρότητας των δεδομένων. Στη περίπτωση που η απάντηση που εισάγεται είναι λάθος το πρόγραμμα να ζητάει ξανά νέα τιμή μέχρι να δοθεί η σωστή.

Μονάδες 8

4.3. Να υπολογίζει και να εμφανίζει το πλήθος των ατόμων που απάντησαν «ΝΑΙ».

Μονάδες 5

4.4. Να υπολογίζει και να εμφανίζει το σύνολο των πολιτών που απάντησαν «ΝΑΙ», όπως επίσης το ποσοστό των ανδρών που απάντησαν «ΝΑΙ» και το ποσοστό των γυναικών που απάντησαν «ΝΑΙ».

Μονάδες 5

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1  i_nai=0 #αρχικοποίηση του μετρητή όλων των απαντήσεων με NAI
2  i_g=0  #αρχικοποίηση του μετρητή όλων των απαντήσεων των γυναικών με NAI
3  i_a=0  #αρχικοποίηση του μετρητή όλων των απαντήσεων των ανδρών με NAI
4  for i in range(0,200):
5      #Δίνουμε το φύλο του πολίτη
6      f=raw_input("Φύλο (Α)Άνδρας ή (Γ)Γυναίκα")
7      #Έλεγχος εγκυρότητας δεδομένων από το πληκτρολόγιο
8      while (f!="Α") and (f!="Γ"):
9          print "Εδωσες λάθος απάντηση..."
10         f=raw_input("Ξαναπροσπάθησε να ξαναδώσεις την απάντηση:")
11     #Δίνουμε την απάντηση του πολίτη
12     ap=raw_input("Χρησιμοποιείται τους κάδους ανακύκλωσης (NAI, ΟΧΙ ή ΔΕΝ ΞΕΡΩ);")
13     #Έλεγχος εγκυρότητας δεδομένων από το πληκτρολόγιο
14     while (ap!="NAI") and (ap!="ΟΧΙ") and (ap!="ΔΕΝ ΞΕΡΩ"):
15         print "Εδωσες λάθος απάντηση..."
16         ap=raw_input("Ξαναπροσπάθησε να ξαναδώσεις την απάντηση:")
17     if ap=="NAI":      #Έλεγχος απάντησης αν είναι "NAI"
18         i_nai=i_nai+1  #Μετράει όλα τα NAI
19         if f=="Γ":      #Έλεγχος απάντησης αν είναι NAI από ΓΥΝΑΙΚΑ
20             i_g=i_g+1  #Μετράει τα NAI των ΓΥΝΑΙΚΩΝ
21         else:
22             i_a=i_a+1  #Μετράει τα NAI των ΑΝΔΡΩΝ
23     p_g=i_g*100/i_nai  #Υπολογισμός ποσοστού γυναικών που απάντησαν NAI
24     p_a=i_a*100/i_nai  #Υπολογισμός ποσοστού ανδρών που απάντησαν NAI
25     #Εκτύπωση αποτελεσμάτων
26     print "Το πλήθος των πολιτών που απάντησαν NAI, είναι:",i_nai
27     print "Το πλήθος των γυναικών που απάντησαν NAI, είναι:",i_g,"με ποσοστό ",p_g,"% "
28     print "Το πλήθος των ανδρών που απάντησαν NAI, είναι:",i_g,"με ποσοστό ",p_a,"% "
```

ΘΕΜΑ 2

2.1. Δίνεται το παρακάτω πρόγραμμα σε γλώσσα Python:

```
1 list=[0,2,4,6,8,10]
2 for x in range(0,4):
3     list[x]=(list[x+1]+list[x+2])/2
4 for x in range(0,6):
5     print list[x]
```

Να γράψετε ποιες τιμές θα εμφανιστούν στην οθόνη κατά την εκτέλεση των εντολών του.

Μονάδες 10

2.2.

A. Δίνεται το παρακάτω πρόγραμμα στο οποίο δίνεται ο βαθμός μαθητή από 0 έως και 20 και τυπώνεται κατάλληλο μήνυμα σύμφωνα με τον παρακάτω πίνακα:

Συνθήκη	Μήνυμα
Βαθμός $\geq$ 18 και Βαθμός $\leq$ 20	Άριστα
Βαθμός $\geq$ 15 και Βαθμός $<$ 18	Αρκετά καλά
Βαθμός $\geq$ 13 και Βαθμός $<$ 15	Σχετικά καλά
Βαθμός $\geq$ 10 και Βαθμός $<$ 13	Καλώς
Βαθμός $<$ 10	Κακώς

Στο πρόγραμμα υπάρχουν τέσσερα (4) κενά , τα οποία έχουν αριθμηθεί και υπογραμμιστεί.

Να γράψετε στο τετράδιο σας τον αριθμό του κάθε κενού και δίπλα την συνθήκη που πρέπει να ικανοποιείται για να εμφανίζεται το αντίστοιχο μήνυμα.


```
1  b=input("Παρακαλώ δώσε το βαθμό του μαθητή: ")
2  if ____ (1) ____:
3      print "Άριστα"
4  else:
5      if ____ (2) ____:
6          print "Άρκετά καλά"
7      else:
8          if ____ (3) ____:
9              print "Σχετικά καλά"
10         else:
11             if ____ (4) ____:
12                 print "Καλώς"
13             else:
14                 print "Κακώς"
```

Μονάδες 8

B.

Γράψτε στο τετράδιο σας ισοδύναμο πρόγραμμα Python με το παραπάνω, κάνοντας χρήση της εντολής if ... elif... αντί της εμφωλευμένης εντολής

```
If ...:
else:
    If ...:
    else:
```

Μονάδες 7

ΘΕΜΑ 2

2.1

5
7
9
8
10

2.2.

A.

1. $b \geq 18$ and $b \leq 20$
2. $b \geq 15$
3. $b \geq 13$
4. $b \geq 10$

B.

```
1  b=input("Παρακαλώ δώσε το βαθμό του μαθητή: ")
2  if b>=18 and b<=20:
3      print "Άριστα"
4  elif b>=15:
5      print "Αρκετά καλά"
6  elif b>=13:
7      print "Σχετικά καλά"
8  elif b>=10:
9      print "Καλώς"
10 else:
11     print "Κακώς"
```

ΘΕΜΑ 4

Δύο καταστήματα βιβλιοπωλείων, κοινής ιδιοκτησίας, έχουν ίδια (με τον ίδιο τίτλο) και διαφορετικά βιβλία (με διαφορετικό τίτλο). Να γράψετε πρόγραμμα σε γλώσσα Python, το οποίο:

4.1. Να διαβάζει τους τίτλους των βιβλίων του 1^{ου} βιβλιοπωλείου. Η είσοδος δεδομένων να σταματά όταν πληκτρολογήσουμε τον χαρακτήρα «*».

Μονάδες 5

4.2. Να διαβάζει τους τίτλους των βιβλίων του 2^{ου} βιβλιοπωλείου. Η είσοδος δεδομένων να σταματά όταν πληκτρολογήσουμε τον χαρακτήρα «*».

Μονάδες 5

4.3. Να εμφανίζει τους τίτλους των βιβλίων που είναι ίδιοι και στα δύο βιβλιοπωλεία.

Μονάδες 10

4.4. Να υπολογίζει και να εμφανίζει το πλήθος των βιβλίων που έχουν τον ίδιο τίτλο και στα δύο βιβλιοπωλεία.

Μονάδες 5

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1  b1=[] #Αρχικοποίηση λίστας 1ου βιβλιοπωλείου
2  #Εισάγουμε τα βιβλία του 1ου βιβλιοπωλείου
3  print "Δώσε τους τίτλους βιβλίου του 1ου καταστήματος"
4  x=raw_input("Δώσε τίτλο βιβλίου (Με *=τέλος πληκτολόγησης):")
5  while x!="*": #Έλεγχος της πληκτρολόγησης
6      b1.append(x)
7      x=raw_input("Δώσε τίτλο βιβλίου (Με *=τέλος πληκτολόγησης):")
8
9  b2=[] #Αρχικοποίηση λίστας 2ου βιβλιοπωλείου
10 #Εισάγουμε τα βιβλία του 2ου βιβλιοπωλείου
11 print "Δώσε τους τίτλους βιβλίου του 2ου καταστήματος"
12 x=raw_input("Δώσε τίτλο βιβλίου(Με *=τέλος πληκτολόγησης) :")
13 while x!="*": #Έλεγχος της πληκτρολόγησης
14     b2.append(x)
15     x=raw_input("Δώσε τίτλο βιβλίου (Με *=τέλος πληκτολόγησης):")
16
17 print "_____ΒΙΒΛΙΑ ΜΕ ΤΟΝ ΙΔΙΟ ΤΙΤΛΟ_____"
18 #Σειριακή αναζήτηση κάθε στοιχείου της 1ης λίστας στη 2η λίστα
19 m=0 #αρχικοποίηση μετρητή
20 for i in range(0,len(b1)): #Οι θέσεις της 1ης λίστας
21     #Αναζήτηση κάθε στοιχείου της 1ης λίστας στη 2η λίστα
22     found=False
23     for j in range(0,len(b2)): #Οι θέσεις της 2ης λίστας
24         if b1[i]==b2[j]: #Έλεγχος για βιβλία με τον ίδιο τίτλο
25             found=True
26             print m+1,".",b1[i] #Εκτύπωση του τίτλου
27     if found==True:
28         #Όταν βρίσκουμε βιβλία με τον ίδιο τίτλο αύξηση μετρητή κατά συν ένα
29         m=m+1
30 #Εκτύπωση πλήθους βιβλίων με τον ίδιο τίτλο
31 print "Βρέθηκαν συνολικά ",m, " ίδια βιβλία"
```

ΘΕΜΑ 2

2.1. Δίνεται το παρακάτω πρόγραμμα σε γλώσσα Python:

```
1  a=[1,2,3,4]
2  b=[1,2,3,4]
3  c=a+b
4  print c
5  d=[0,0,0,0]
6  for i in range(0,4):
7      d[i]=a[i]+b[i]
8  print d
```

Να γράψετε ποιες τιμές θα εμφανιστούν στην οθόνη κατά την εκτέλεση των εντολών του.

Μονάδες 12

2.2.

Δίνεται το παρακάτω πρόγραμμα σε γλώσσα προγραμματισμού Python:

```
1  i=10
2  while i<=100:
3      print i
4      i=i+10
```

A. Γράψτε στο γραπτό σας τι εμφανίζεται στην οθόνη του υπολογιστή κατά την εκτέλεση του προγράμματος.

Μονάδες 10

B. Να γράψτε ισοδύναμο πρόγραμμα σε Python, χρησιμοποιώντας την εντολή επανάληψης for αντί της της εντολής επανάληψης while.

Μονάδες 3

QEMA 2

2.1

[1, 2, 3, 4, 1, 2, 3, 4]

[2, 4, 6, 8]

2.2.

A.

10
20
30
40
50
60
70
80
90
100

B.

```
1  for i in range(10,101,10):  
2      print i
```

ΘΕΜΑ 4

Σε ένα γκάλοπ συμμετείχαν 150 πολίτες και απάντησαν στην ερώτηση :

«Τι θα ψηφίσετε στις επόμενες εκλογές

K1=ΚΟΜΜΑ_1, K2=ΚΟΜΜΑ_2, K3=ΚΟΜΜΑ_3 ή K4=ΚΟΜΜΑ_4

Δώσε επιλογή:»

Να γράψετε πρόγραμμα σε γλώσσα Python το οποίο:

4.1. Να διαβάζει τις απαντήσεις των πολιτών έτσι, ώστε όπου ΚΟΜΜΑ_1 πληκτρολογούν Κ1, όπου ΚΟΜΜΑ_2 πληκτρολογούν Κ2, όπου ΚΟΜΜΑ_3 πληκτρολογούν Κ3 και όπου ΚΟΜΜΑ_4 πληκτρολογούν Κ4. Τις απαντήσεις τις αποθηκεύουν σε μία λίστα με το όνομα Κ. Να θεωρήσετε ότι τα δεδομένα δίνονται σωστά και δεν χρειάζεται οποιοσδήποτε έλεγχος.

Μονάδες 5

4.2. Να υπολογίζει και να εμφανίζει το όνομα του κάθε κόμματος και το πλήθος των απαντήσεων που πήρε κάθε κόμμα.

Μονάδες 10

4.3. Να υπολογίζει και να εμφανίζει το κόμμα με τις περισσότερες προτιμήσεις, εμφανίζοντας το μήνυμα: «Δημοφιλέστερο κόμμα το <όνομα κόμματος>.»

Μονάδες 10

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1  k=[]  #Αρχικοποίηση λίστας
2  # Δίνω τις τιμές από το πληκτρολόγιο και τις καταχωρώ στην λίστα
3  for i in range(0,150):
4      print "Τι θα ψηφίσετε στις επόμενες εκλογές "
5      print "(K1=KOMMA_1, K2=KOMMA_2, K3=KOMMA_3 ή
6      K4=KOMMA_4"
7      x=raw_input("Δώσε επιλογή: ")
8      k.append(x)
9
10 # Λίστα με τα ονόματα των κομμάτων
11 komma=["KOMMA_1","KOMMA_2","KOMMA_3","KOMMA_4"]
12 #Λίστα με το όνομα κόμματος στην απάντηση
13 on_k=["K1","K2","K3","K4"]
14 m=[0,0,0,0]  #Αρχικοποίηση της λίστας μετρητής
15
16 for i in range(0,4):  # Οι θέσεις των 4 κομμάτων
17     for j in range(0,150): # Οι θέσεις των απαντήσεων
18         if on_k[i]==k[j]: #Έλεγχος της κάθε απάντησης σε ποιο κόμμα είναι
19             m[i]=m[i]+1  # Αύξηση του μετρητή για κάθε κόμμα
20
21 for i in range(0,4):
22     print komma[i],m[i]  # Εκτύπωση όλων κομμάτων με το πλήθος επιλογών του
23
24 #Εύρεση του κόμματος με τους περισσότερους ψήφους
25 max=m[0]  #Αρχικοποίηση του max
26 thesi=0  #Αρχικοποίηση της μεταβλητής που δίνει τη θέση του μεγαλύτερου
27 for i in range(1,4):
28     if m[i]>max:
29         max=m[i]
30         thesi=i
31
32 #Εκτύπωση του κόμματος με τους περισσότερους ψήφους
33 print "Δημοφιλέστερο κόμμα είναι το ", komma[thesi]
```


ΘΕΜΑ 2

2.1. Να γράψετε στο τετράδιο τον αριθμό που βρίσκεται στην αριστερή στήλη και δίπλα του πόσες φορές εκτελείτε η κάθε μια από τις επαναλήψεις:

1.	<pre>1 for i in range(3,3): 2 print i</pre>
2.	<pre>1 for i in range(10,5): 2 print i</pre>
3.	<pre>1 for i in range(3,0,-1): 2 print i</pre>
4.	<pre>1 for i in range(5,10,2): 2 print i</pre>
5.	<pre>1 for i in range(100,10,-20): 2 print i</pre>

Μονάδες 10

2.2. Δίνεται το παρακάτω πρόγραμμα σε γλώσσα Python, στο οποίο πληκτρολογούμε το κωδικό συνδρομητή και το χρόνο ομιλίας του. Το πρόγραμμα υπολογίζει και τυπώνει την μηνιαία χρέωση του συνδρομητή κινητής τηλεφωνίας. Η εταιρεία της κινητής τηλεφωνίας ακολουθεί την παρακάτω πολιτική τιμών, για την χρέωση των συνδρομητών της. Η χρέωση γίνεται κλιμακωτά.

Πάγιο 8€	
Χρόνος σε δευτερόλεπτα	Χρονοχρέωση (€/δευτερόλεπτο)
Από 1 έως και 300	0.098
Από 301 έως και 1.000	0.07
Από 1.001 και άνω	0.04

Η παραπάνω διαδικασία θα επαναλαμβάνεται μέχρι να δώσουμε τον χαρακτήρα «*» στο όνομα του συνδρομητή.

Στο πρόγραμμα υπάρχουν πέντε (5) κενά, τα οποία είναι αριθμημένα και υπογεγραμμένα. Να γράψετε στο τετράδιο σας τον αριθμό του κάθε κενού και δίπλα τι πρέπει να συμπληρωθεί, ώστε το πρόγραμμα να εκτελεί τη λειτουργία που επιγράφηκε.

```
1  o=raw_input("Δώσε κωδικό συνδρομητή: ")
2  while __ (1)__:
3      xr=input("Δώσε χρονική διάρκεια τηλεφωνιμάτων: ")
4      if __ (2)__:
5          xrewsi=xr*0.098
6      elif __ (3)__:
7          xrewsi=300*0.98+__ (4)__ *0.07
8      else:
9          xrewsi=300*0.98+700*0.7+__ (5)__ *0.04
10     xrewsi=8+xrewsi
11     print "Η μηνιαία χρέωση του συνδρομητή είναι: ", xrewsi, "€"
```

Μονάδες 15

ΘΕΜΑ 2

2.1

1. Καμία
2. Καμία
3. 3 φορές
4. 3 φορές
5. 5 φορές

2.2.

1. $o \neq "*"$
2. $xr \geq 1$ and $xr \leq 300$
3. $xr \leq 1000$
4. $(xr - 300)$
5. $(xr - 1000)$

ΘΕΜΑ 4

Ένα Λύκειο θέλει να έχει στατιστικά στοιχεία για τους τελειόφοιτους μαθητές, σε σχέση με τον βαθμό απολυτηρίου. Να γράψετε πρόγραμμα στη γλώσσα προγραμματισμού Python, το οποίο:

4.1. Να διαβάζει το ονοματεπώνυμο και το βαθμό απολυτηρίου των αποφοίτων. Η πληκτρολόγηση σταματά όταν στο όνομα δώσουμε τον χαρακτήρα «*». Το πρόγραμμα να κάνει έλεγχο για το βαθμό, ώστε να είναι στο διάστημα [10,20]. Στην περίπτωση που δοθεί λανθασμένος βαθμός να εμφανίζεται το μήνυμα «Άκυρος βαθμός » και το πρόγραμμα να ζητά να δοθεί ο βαθμός, μέχρι να δοθεί έγκυρη τιμή.

Μονάδες 6

4.2. Να υπολογίζει και να εμφανίζει το πλήθος των αποφοίτων.

Μονάδες 4

4.3. Να υπολογίζει και να εμφανίζει το πλήθος και το ποσοστό των αποφοίτων με βαθμό απολυτηρίου μεγαλύτερο του 18.

Μονάδες 5

4.4. Να υπολογίζει και να εμφανίζει το πλήθος και το ποσοστό των αποφοίτων με βαθμό απολυτηρίου μικρότερο ή ίσο του 18 και μεγαλύτερο του 15.

Μονάδες 5

4.5. Να υπολογίζει και να εμφανίζει το πλήθος και το ποσοστό των αποφοίτων με βαθμό απολυτηρίου μικρότερο ή ίσο του 15 και μεγαλύτερο ή ίσο του 10.

Μονάδες 5

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1  #Αρχικοποίηση μετρητών
2  pl=0  #Μετρητής πλήθους αποφοίτων
3  m18=0  # Μετρητής βαθμών>18
4  m15=0  #Μετρητής βαθμών με 15<βαθμό<=18
5  m10=0  #Μετρητής βαθμών με 10<=βαθμό<=15
6
7  onoma=raw_input("Δώσε ονοματεπώνυμο απόφοιτου: ")
8  while onoma!="*":  #Επαναληπτική διαδικασία μέχρι να δώσουμε * στο ονομ/μο
9      bathmos=input("Δώσε τον βαθμό απολυτηρίου του αποφοίτου: ")
10     while bathmos<10 or bathmos>20:  #Έλεγχος πληκτρολογίου για το βαθμό
11         print "Άκυρος βαθμός"
12         bathmos=input("Δώσε τον βαθμό απολυτηρίου του αποφοίτου: ")
13     pl=pl+1      #Μέτρηση αποφοίτου
14     if bathmos>18:
15         m18=m18+1  #Μέτρηση με βαθμό>18
16     elif bathmos>15:
17         m15=m15+1  #Μέτρηση με βαθμό>18
18     elif bathmos>=10:
19         m10=m10+1  #Μέτρηση με βαθμό>18
20     onoma=raw_input("Δώσε ονοματεπώνυμο απόφοιτου: ")
21     print "_____ "
22     if pl<>0:
23         #Εύρεση ποσοστών
24         p18=m18*100/pl
25         p15=m15*100/pl
26         p10=m10*100/pl
27         #Εμφάνιση αποτελεσμάτων
28         print "Σύνολο αποφοίτων: ",pl
29         print "Απόφοιτοι με βαθμό απολυτηρίου>18: ",m18, "ποσοστό ",p18,"% "
30         print "Απόφοιτοι με 18<=βαθμό απολυτηρίου>15: ",m15,"ποσοστό ",p15,"% "
31         print "Απόφοιτοι με 15<=βαθμό απολυτηρίου<=10: ",m10,"ποσοστό ",p10,"% "
32     else:
33         print "Δεν δόθηκαν απόφοιτοι."
```

ΘΕΜΑ 2

2.1. Να γράψετε στο γραπτό σας ισοδύναμο με το παρακάτω τμήμα προγράμματος Python, στο οποίο η σύνθετη if θα έχει αντικατασταθεί από δύο απλές if:

```
1   if bathmos>=5 and bathmos <=10:
2       print "ΕΓΚΥΡΟΣ ΒΑΘΜΟΣ ΠΤΥΧΙΟΥ"
3   else:
4       print "ΑΚΥΡΟΣ ΒΑΘΜΟΣ ΠΤΥΧΙΟΥ"
```

Μονάδες 10

2.2. Δίνεται το πιο κάτω πρόγραμμα σε γλώσσα προγραμματισμού Python, το οποίο με τη βοήθεια της συνάρτησης **ypotinousa** υπολογίζει το μήκος της υποτείνουσας ενός ορθογωνίου τριγώνου, με δεδομένα εισόδου τα μήκη των δύο κάθετων πλευρών του.

```
import math
```

A

```
athr_tetr= pow(a,2)+ pow(b,2)
ipot=math.sqrt( athr_tetr )
return ipot
```

```
plevra1=float(input("Δώσε το μήκος της 1ης κάθετου πλευράς του ορθογωνίου τριγώνου:"))
plevra2=float(input("Δώσε το μήκος της 2ης κάθετου πλευράς:"))
print ypotinousa(plevra1,plevra2)
```

Να γράψετε στο γραπτό σας την επικεφαλίδα της συνάρτησης **ypotinousa**, που πρέπει να τοποθετηθεί στη θέση A, έτσι ώστε η συνάρτηση να δέχεται τα μήκη των δύο κάθετων πλευρών ενός ορθογωνίου τριγώνου και να υπολογίζει και να επιστρέφει στο κυρίως πρόγραμμα το μήκος της υποτείνουσάς του.

Μονάδες 6

2.3. Δίνεται το πιο κάτω πρόγραμμα στη γλώσσα προγραμματισμού Python:

```
1  def add1(a, b):  
2      a = a + 1  
3      b = b + 1  
4      return a+b  
5  # τέλος συνάρτησης  
6  x = 0  
7  y = 1  
8  z = add1( y, x )  
9  print x, y, z
```

Τι θα εμφανίσει το παραπάνω πρόγραμμα;

Μονάδες 9

ΘΕΜΑ 2

2.1 Το ισοδύναμο τμήμα προγράμματος είναι το ακόλουθο:

```
1   if bathmos>=5 and bathmos <=10:  
2       print "ΕΓΚΥΡΟΣ ΒΑΘΜΟΣ ΠΤΥΧΙΟΥ"  
3   if bathmos<5 or bathmos >10:  
4       print "ΑΚΥΡΟΣ ΒΑΘΜΟΣ ΠΤΥΧΙΟΥ"
```

2.2

Η επικεφαλίδα της συνάρτησης ***ypotinousa*** που πρέπει να τοποθετηθεί στο κενό Α είναι :

def ypotinousa(a, b):

2.3 Κατά την εκτέλεση του δοσμένου προγράμματος Python, θα εμφανιστούν στην οθόνη τα ακόλουθα:

0 1 3

ΘΕΜΑ 4

Ένας μαθητής έχει μια συλλογή από συλλεκτικούς δίσκους βινυλίου που κυκλοφόρησαν μεταξύ του 1950 και του 1980, συμπεριλαμβανομένων. Να γράψετε πρόγραμμα σε Python το οποίο:

4.1. Για κάθε δίσκο, να διαβάζει τον τίτλο του, το όνομα του καλλιτέχνη ή του συγκροτήματος και το έτος κυκλοφορίας του δίσκου, ελέγχοντας ότι το έτος κυκλοφορίας ανήκει στο διάστημα [1950, 1980]. Στην περίπτωση που εισάγεται έτος εκτός των επιτρεπόμενων τιμών, να εμφανίζεται μήνυμα λάθους και το πρόγραμμα να ζητά ξανά το έτος κυκλοφορίας, μέχρι να δοθεί έγκυρη τιμή. Τα στοιχεία των δίσκων θα καταχωρίζονται στις λίστες TITLES, ARTISTS και YEARS, αντίστοιχα.

Μονάδες 7

4.2. Η εισαγωγή των δεδομένων ολοκληρώνεται όταν δοθεί ως τίτλος ο χαρακτήρας '-'. Θεωρήστε ότι θα καταχωρηθούν τα στοιχεία ενός τουλάχιστον δίσκου.

Μονάδες 4

4.3. Θα βρίσκει και θα εμφανίζει το πλήθος των δίσκων της συλλογής που κυκλοφόρησαν μετά το 1970.

Μονάδες 4

4.4. Στη συνέχεια, θα ζητάει από το χρήστη το όνομα ενός καλλιτέχνη/συγκροτήματος και θα εμφανίζει τα στοιχεία των δίσκων αυτού του καλλιτέχνη/ συγκροτήματος που περιλαμβάνει η συλλογή, σε χωριστή γραμμή για κάθε δίσκο, ως εξής: Τίτλος δίσκου (έτος κυκλοφορίας). Για παράδειγμα: Shades of Deep Purple (1968). Αν δεν περιλαμβάνει η συλλογή δίσκους αυτού του καλλιτέχνη/συγκροτήματος, το πρόγραμμα θα πρέπει να εμφανίζει το μήνυμα «Δε βρέθηκαν δίσκοι !».

Μονάδες 10

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1  # 4.3
2  pl= 0 # αρχικοποίηση πλήθους δίσκων που κυκλοφόρησαν μετά το 1970
3
4  # Δημιουργία των λιστών για τα στοιχεία των δίσκων
5  TITLES= []
6  ARTISTS= []
7  YEARS= []
8
9  # 4.1 Εισαγωγή τίτλου πρώτου δίσκου
10 title=raw_input("Δώσε τον τίτλο του πρώτου δίσκου:")
11 # Ερώτημα 4.2
12 while title != '-':
13     # 4.1 Εισαγωγή στοιχείων δίσκου
14     artist=raw_input("Δώσε τον καλλιτέχνη/συγκρότημα:")
15     year=int(input("Δώσε το έτος κυκλοφορίας [1950-1980]:"))
16     # έλεγχος εγκυρότητας για το έτος κυκλοφορίας
17     while year <1950 or year>1980:
18         print "'Ακυρη τιμή !"
19         year=int(input("Δώσε ξανά το έτος [1950-1980]:"))
20     # 4.1 Καταχώρηση στοιχείων δίσκου στις αντίστοιχες λίστες
21     TITLES.append(title) # πρόσθεσε τον τίτλο στο τέλος της λίστας TITLES
22     ARTISTS.append(artist) # πρόσθεσε τον καλλιτέχνη στο τέλος της λίστας ARTISTS
23     YEARS.append(year) # πρόσθεσε το έτος στο τέλος της λίστας YEARS
24     # 4.3
25     if year > 1970:
26         pl = pl +1
27     # 4.2
28     title=raw_input("Δώσε τον τίτλο του επόμενου δίσκου ή - για τερματισμό:")
29
30 # 4.3
31 print "Πλήθος δίσκων της συλλογής που κυκλοφόρησαν μετά το 1970: ", pl
32 # 4.4
33 name= raw_input("Δώσε όνομα καλλιτέχνη/ συγκροτήματος προς αναζήτηση:")
34 found= False
35 for i in range(len(ARTISTS)): # σειριακή αναζήτηση
36     if ARTISTS[ i ] == name :
37         print TITLES[ i ]+ " (" + str(YEARS[ i ])+ ")"
38         found= True # βρήκαμε τουλάχιστον ένα δίσκο του καλλιτέχνη/συγκροτήματος
39 if found == False:
40     print "Δε βρέθηκαν δίσκοι !"
```


ΘΕΜΑ 2

2.1. Να συμπληρώσετε κατάλληλα την παρακάτω συνθήκη (**εντολή 2**) έτσι ώστε να εξασφαλίζεται ότι ο χρήστης θα εισάγει τελικά έναν βαθμό που θα είναι στο διάστημα [1-20], σε διαφορετική περίπτωση θα εμφανίζεται μήνυμα λάθους.

```
1  vathmos=int(input('Δώστε το βαθμό του/της μαθητή/τριας: '))
2  while _____ :
3      vathmos=int(input('Δώσατε λάθος βαθμό. Δώστε ξανά το βαθμό [1-20]: '))
```

Μονάδες 3

2.2. Δίνεται ο παρακάτω κώδικας σε γλώσσα προγραμματισμού Python.

```
1  γ,x=5,3
2  s=0
3  for i in range (γ):
4      s=s+x
5      print i
6  print s
```

Επίσης δίνεται το παρακάτω υπόδειγμα πίνακα (πίνακας τιμών), όπου έχει συμπληρωθεί η εκτέλεση της πρώτης εντολής του προγράμματος.

	x	y	s	i	Έξοδος
Αρχικές Τιμές	3	5	0	?	
1 ^η επανάληψη	...	...	...	...	
....	...	...	...	...	...

Να μεταφέρεται στο γραπτό σας τον παραπάνω πίνακα και να τον συμπληρώσετε με τις τιμές που παίρνουν οι μεταβλητές κατά την εκτέλεση του προγράμματος. Να προσθέσετε στον πίνακα όσες γραμμές είναι απαραίτητες.

Μονάδες 12

2.3. Να μεταφέρετε στο γραπτό σας και να συμπληρώσετε τα κενά στο παρακάτω πρόγραμμα ώστε αυτό να υπολογίζει και να εμφανίζει το άθροισμα της παρακάτω αριθμητικής παράστασης:

$$s = 1^2 + 2^3 + 3^4 + 4^5 + 5^6$$

```
1  s = _____
2  for i in range ( _____ , _____ ) :
3      s = _____ + _____
4  print s
```

Μονάδες 10

ΕΝΔΕΙΚΤΙΚΕΣ ΛΥΣΕΙΣ

ΘΕΜΑ 2

2.1. 2. **while** vathmos<1 or vathmos>20 :

2.2.

	x	y	s	i	Έξοδος
Αρχικές Τιμές	3	5	0	?	
1 ^η επανάληψη			3	0	0
2 ^η επανάληψη			6	1	1
3 ^η επανάληψη			9	2	2
4 ^η επανάληψη			12	3	3
5 ^η επανάληψη			15	4	4
					15

2.3.

```
1 s = 0
2 for i in range ( 1 , 6 ) :
3     s = s + (i**(i+1))
4 print s
```

ΘΕΜΑ 4

Το κάθε βαγόνι του τελεφερίκ στο λόφο του Λυκαβηττού έχει μέγιστη χωρητικότητα 34 άτομα και μέγιστο όριο ασφαλείας βάρους τα 4500 κιλά. Να γράψετε πρόγραμμα σε Python το οποίο:

4.1. Θα ενημερώνει συνεχώς με κατάλληλα μηνύματα εάν επιτρέπεται να εισέλθει κάποιο άτομο στο βαγόνι και θα διαβάζει το βάρος του σε κιλά. Η διαδικασία εισαγωγής των δεδομένων ολοκληρώνεται είτε όταν το τελεφερίκ γεμίσει (σε σχέση με τη μέγιστη χωρητικότητα) είτε ξεπεραστεί το όριο ασφαλείας βάρους (σε σχέση με το συνολικό βάρος) είτε όταν δοθεί ως βάρος η τιμή 0, οπότε και θα ξεκινάει το τελεφερίκ.

Μονάδες 8

Κατά την διάρκεια της διαδρομής του τελεφερίκ στο λόφο του Λυκαβηττού θα υπολογίζει και θα εμφανίζει:

4.2. Την μέση τιμή βάρους και το πλήθος των ατόμων που εισήλθαν τελικά στο τελεφερίκ.

Μονάδες 4

4.3. Ποιο είναι το μεγαλύτερο βάρος του ατόμου που εισήλθε στο τελεφερίκ.

Μονάδες 7

4.4. Το ποσοστό των ατόμων που είχαν βάρος πάνω από 100 κιλά.

6 Μονάδες

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΕΝΔΕΙΚΤΙΚΕΣ ΛΥΣΕΙΣ

ΘΕΜΑ 4

```
1  pla=0
2  max_varos=-1
3  sum_varous=0
4  p100=0
5  safe_orio=True
6  while safe_orio==True: #4.1 Εάν επιτρέπεται να εισέλθει κάποιος
7      varos=int(input('Δωσε το βάρος του ατόμου που θέλει να εισέλθει '))
8      if varos==0: #4.1 Εάν δοθεί βάρος =0 να σταματήσει
9          safe_orio=False
10     else:
11         #4.1 Εάν δεν έχει ξεπεραστεί το όριο ασφαλείας βάρους
12         if sum_varous+varos<=4500: #4.1 Όριο ασφαλείας συνολικού βάρους
13             pla=pla+1 # 4.2 πλήθος ατόμων που εισήλθαν
14             sum_varous=sum_varous+varos # 4.2 συνολικό βάρος εισηλθέντων
15             #4.1 Ενημερωτικό μήνυμα για τρέχουσα κατάσταση
16             print 'μπορούν να μπουν ακόμη ',35-pla,' άτομα'
17             print 'Το συνολικό βάρος είναι',sum_varous
18             if pla==35: #4.1 'Εαν έφτασε το όριο ασφαλείας (χωρητικότητα)
19                 safe_orio=False
20         else:
21             safe_orio=False
22
23     if varos>100: #4.4 πλήθος ατόμων με βάρος>100 κιλά
24         p100=p100+1
25
26     if varos>max_varos: #4.3 εύρεση μέγιστου βάρους εισερχόμενου ατόμου
27         max_varos=varos
28
29 if pla==0:
30     print 'Δεν εισήλθε κανένα άτομο '
31 else:
32     # 4.2 Υπολογισμός μέσου όρου βάρους και πόσα άτομα εισήλθαν
33     meso_varos=sum_varous/float(pla)
34     print 'Το μέσο βάρος είναι',meso_varos,' κιλά και εισήλθαν ',pla,' άτομα'
35     #4.3 Εμφάνιση μέγιστου βάρους
36     print 'Το πιο βαρύ άτομο που μπήκε στο τελεφερίκ είναι ',max_varos,' κιλά'
37 #4.4 Υπολογισμός ποσοστού ατόμων που εισήλθαν με βάρος>100 κιλά
38 if p100!=0:
39     pososto=p100/float(pla)*100
40     print 'Το ποσοστό πάνω από 100 κιλά είναι ',pososto,' %'
41 else:
42     print 'Δεν υπήρξαν άτομα πάνω από 100 κιλά'
```


ΘΕΜΑ 2

2.1. Να γράψετε ένα τμήμα προγράμματος σε Python το οποίο θα εμφανίζει τους παρακάτω αριθμούς:

11111

2222

333

44

5

Υπόδειξη: θα πρέπει στην υλοποίηση του κώδικα προγράμματος σε Python να χρησιμοποιηθεί δομή επανάληψης τουλάχιστον μια φορά:

Μονάδες 13

2.2 Δίνεται το παρακάτω τμήμα προγράμματος:

```
L=[4,9,2,3,7,-5]
```

```
1 print len(L)
```

```
L=[12]+L
```

```
2 print L
```

```
L.append(6)
```

```
3 print L
```

```
L.pop()
```

```
4 print L
```

```
L.pop(-2)
```

```
5 print L
```

```
L.insert(3,0)
```

```
6 print L
```

Να γράψετε στο γραπτό σας τους αριθμούς **1, 2, 3, 4, 5** και **6** που αντιστοιχούν στις εντολές print του παραπάνω τμήματος προγράμματος και δίπλα σε κάθε αριθμό το αποτέλεσμα που θα εμφανιστεί στην οθόνη κατά την εκτέλεσή του.

Μονάδες 6

2.3 Δίνεται το παρακάτω τμήμα προγράμματος σε γλώσσα προγραμματισμού Python, το οποίο υπολογίζει και τυπώνει την δύναμη ενός αριθμού, **χωρίς να γίνεται η χρήση του τελεστή ** και της ενσωματωμένης συνάρτησης pow.**

Πρόγραμμα υπολογισμού δύναμης

basi=int(input('Δώσε έναν αριθμό '))

ekth=int(input('Δώσε και την δύναμη του '))

a = (1)

for i **in** range((2)):

a = a* (3)

print 'Το αποτέλεσμα είναι ', a

Στο τμήμα προγράμματος υπάρχουν υπογραμμισμένα κενά τα οποία έχουν αριθμηθεί. Να γράψετε στο τετράδιό σας τους αριθμούς **1, 2, και 3** που αντιστοιχούν στα κενά του παραπάνω προγράμματος και δίπλα σε κάθε αριθμό αυτό που πρέπει να συμπληρωθεί για να υλοποιείται σωστά το πρόγραμμα.

Μονάδες 6

ΘΕΜΑ 2

2.1.

```
1 k=1 #Η τιμή που θέλουμε να εμφανιστεί
2 for i in range(5,0,-1): #Οι γραμμές που θα εμφανιστούν
3     print str(k)*i #πόσες φορές θέλουμε να εμφανιστεί ο αριθμός
4     k=k+1 #επόμενος αριθμός
```

2.2.

1. 6
2. [12, 4, 9, 2, 3, 7, -5]
3. [12, 4, 9, 2, 3, 7, -5, 6]
4. [12, 4, 9, 2, 3, 7, -5]
5. [12, 4, 9, 2, 3, -5]
6. [12, 4, 9, 0, 2, 3, -5]

2.3.

1. 1
2. ekth
3. basi

ΘΕΜΑ 4

Ένας ραδιοφωνικός σταθμός έχει μια ψυχαγωγική εκπομπή η οποία εκπέμπει καθημερινά όλες τις ημέρες της εβδομάδας την ίδια ώρα. Η διοίκηση του σταθμού θέλει να ελέγξει την ακροαματικότητα της εκπομπής αυτής και ζήτησε από μια εταιρεία που ασχολείται με τις ακροαματικότητες των ραδιοφωνικών σταθμών τις μετρήσεις μιας εβδομάδας προκειμένου να εξάγει κάποια συμπεράσματα. Να φτιάξετε πρόγραμμα σε γλώσσα προγραμματισμού Python το οποίο:

4.1. Να διαβάζει το πλήθος των ακροατών που άκουσαν την συγκεκριμένη εκπομπή κάθε ημέρα της εβδομάδας. Υποθέστε ότι η 1^η ημέρα είναι η ΔΕΥΤΕΡΑ και η 7^η η ΚΥΡΙΑΚΗ.

Μονάδες 4

4.2. Υπολογίζει και εκτυπώνει αν η μέση τιμή ακροαματικότητας των καθημερινών εκπομπών είναι η ίδια με του Σαββατοκύριακου ή ποια είναι υψηλότερη μετά από κατάλληλα διαμορφωμένο μήνυμα.

Μονάδες 12

4.3. Υπολογίζει και εμφανίζει την υψηλότερη ακρόαση και ποια ημέρα της εβδομάδας έγινε αυτή.

Μονάδες 4

4.4. Ελέγχει και ενημερώνει με κατάλληλο μήνυμα, εάν άκουσαν την εκπομπή παραπάνω από 500 ακροατές τουλάχιστον τις μισές ημέρες της εβδομάδας.

Μονάδες 5

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
    print b
```

ΘΕΜΑ 4

```
1  s_wek=0 # 4.2 σύνολο σαββατοκύριακου
2  pl=0   # 4.4 πλήθος ακροατών>500
3  meg_ak=-1 #Μέγιστη ακρόαση
4  for i in range (1,8):
5      # 4.1 Εισαγωγή δεδομένων για καθημερινές
6      if i<=5:
7          ak=int(input('Δώσε αριθμό ακροατών της '+str(i)+'ης ημέρας'))
8          sd=sd+ak #4.2 Συνολικό άθροισμα ακροατών τις καθημερινές
9      # 4.1 Εισαγωγή δεδομένων για Σαββατοκύριακο
10     if i>=6 and i<=7:
11         ak=int(input('Δώσε αριθμό ακροατών της '+str(i)+'ης ημέρας'))
12         s_wek=s_wek+ak #4.2 Συνολικό άθροισμα ακροατών το Σαββατοκύριακο
13     #4.3 Υπολογισμός μέγιστης ακρόασης
14     if ak>meg_ak:
15         meg_ak=ak
16         imera=i    #ποιά ημέρα ήταν
17     #4.4 Πλήθος ακροατών >500
18     if ak>500:
19         pl=pl+1
20 #4.2 Υπολογισμός μέσων τιμών ακρόασης
21 mo_d=sd/5.    #Μέσος Όρος καθημερινών
22 mo_wek=s_wek/2. #Μέσος Όρος Σαββατοκύριακου
23 #4.2 Έλεγχος πότε υπήρξε μεγαλύτερη ακρόαση
24 if mo_d>mo_wek:
25     print 'Μεγαλύτερη ακροαματικότητα τις καθημερινές'
26 elif mo_d<mo_wek:
27     print 'Μεγαλύτερη ακροαματικότητα το Σαββατοκύριακο'
28 else:
29     print 'Ίδια ακροαματικότητα τις καθημερινές και το Σαββατοκύριακο'
30 #4.3 Εμφάνιση της ημέρας που έγινε η μεγαλύτερη ακρόαση
31 print 'Την ',imera,'η ημέρα είχε την μεγαλύτερη ακροαματικότητα με ',meg_ak,' ακροατές'
32 #4.4 Έλεγχος αν υπήρχαν τις μισές ημέρες της εβδομάδος >500 ακροάσεις
33 if pl>7/2.:
34     print 'Άκουσαν την εκπομπή παραπάνω από 500 ακροατές τις μισές',
35     print 'ημέρες της εβδομάδας'
36 else:
37     print 'Δυστυχώς δε άκουσαν την εκπομπή παραπάνω από 500 ακροατές',
38     print 'τις μισές ημέρες της εβδομάδας'
```

ΘΕΜΑ 2

2.1. Να γράψετε στο γραπτό σας τι θα εμφανίσει στην οθόνη το παρακάτω πρόγραμμα κατά την εκτέλεση του.

```
1  L=[4, 6, 1, 7]
2  for i in L:
3      print i
4      if i % 3 == 0:
5          L.append( i-2 )
```

Μονάδες 10

2.2. Δίνεται το παρακάτω πρόγραμμα σε γλώσσα προγραμματισμού Python, το οποίο υπολογίζει το άθροισμα $S = 5^2 + 10^2 + 15^2 + \dots$, μέχρι αυτό να ξεπεράσει την τιμή 1000.

Το πρόγραμμα έχει πέντε αριθμημένα κενά. Να γράψετε στο γραπτό σας τον αριθμό κάθε κενού (1-5) και δίπλα τι πρέπει να συμπληρωθεί στο κενό αυτό, ώστε το πρόγραμμα να εκτελεί την προβλεπόμενη λειτουργία.

```
1  S = __(1)___
2  x = __(2)___
3  while S __(3)__:
4      S = __(4)___
5      x = __(5)___
6  print S
```

Μονάδες 15

ΘΕΜΑ 2

2.1 Οι τιμές που εμφανίζονται στην οθόνη κατά την εκτέλεση του προγράμματος είναι οι εξής:

4

6

1

7

4

2.2

(1): 0

(2): 5

(3): ≤ 1000

(4): $S + x * x$

(5): $x + 5$

ΘΕΜΑ 4

Μια ελληνική εταιρεία κατασκευής γυναικείων υποδημάτων θέλει να επεκτείνει τις δραστηριότητές της στη Μ. Βρετανία, ανοίγοντας ένα κατάστημα στο κέντρο του Λονδίνου. Έχει επιλέξει για το κατάστημα αυτό κάποια επωνυμία, αλλά πρέπει να ερευνήσει αν η συγκεκριμένη επωνυμία έχει ήδη δοθεί σε κάποιο από τα 3000 καταστήματα υποδημάτων που λειτουργούν στην περιοχή του Λονδίνου. Να γράψετε πρόγραμμα σε Python το οποίο:

4.1. Για κάθε ένα από τα 3000 καταστήματα υποδημάτων που λειτουργούν στο Λονδίνο διαβάζει την επωνυμία του και το ονοματεπώνυμο του ιδιοκτήτη του (θεωρείστε πως κάθε κατάστημα έχει έναν μόνο ιδιοκτήτη). Τα στοιχεία των 3000 καταστημάτων θα καταχωρίζονται στις λίστες NAMES και OWNERS, αντίστοιχα. Θεωρείστε ότι όλες οι επωνυμίες που καταχωρούνται είναι μοναδικές και όλα τα δεδομένα εισάγονται με κεφαλαία λατινικά γράμματα. Στη συνέχεια, διαβάζει την επιθυμητή επωνυμία για το νέο κατάστημα.

Μονάδες 10

4.2. Βρίσκει, με τη βοήθεια συνάρτησης την οποία θα υλοποιήσετε, αν είναι διαθέσιμη η επιθυμητή επωνυμία. Η συνάρτηση αυτή θα δέχεται σαν παραμέτρους την επιθυμητή επωνυμία και τις λίστες NAMES και OWNERS και θα επιστρέφει την τιμή "FREE", αν δεν χρησιμοποιείται η επωνυμία. Διαφορετικά, θα επιστρέφει το ονοματεπώνυμο του ιδιοκτήτη του καταστήματος που λειτουργεί με την επωνυμία αυτή. Το κυρίως πρόγραμμα θα εμφανίζει «ΔΙΑΘΕΣΙΜΗ ΕΠΩΝΥΜΙΑ» ή «ΔΕΣΜΕΥΜΕΝΗ ΑΠΟ » και το ονοματεπώνυμο του ιδιοκτήτη, αντίστοιχα.

Μονάδες 15

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
    print b
```


ΘΕΜΑ 4

```
1  # 4.2
2  def check_name(storename, NAMES, OWNERS):
3      found=False
4      for i in range(len(NAMES)):
5          if NAMES[ i ] == storename : # βρήκαμε την επωνυμία
6              pos=i      # κρατάμε τη θέση της στη λίστα NAMES
7              found= True
8      if found== True:
9          return OWNERS[pos]
10     else:
11         return "FREE"
12
13 # Δημιουργία των λιστών για τα στοιχεία των καταστημάτων
14 TITLES= []
15 OWNERS= []
16 pl_stores=3000 # πλήθος καταστημάτων
17
18 for i in range(pl_stores):
19     # 4.1 Εισαγωγή στοιχείων καταστήματος
20     store=raw_input("Δώσε την επωνυμία του καταστήματος (κεφαλαία λατινικά):")
21     owner=raw_input("Δώσε το ονοματεπώνυμο του ιδιοκτήτη:")
22
23     # 4.1 Καταχώρηση στοιχείων καταστήματος στις αντίστοιχες λίστες
24     TITLES.append(store)      # πρόσθεσε την επωνυμία στο τέλος της λίστας TITLES
25     OWNERS.append(owner) # πρόσθεσε τον ιδιοκτήτη στο τέλος της λίστας OWNERS
26
27 # 4.1
28 desired_name=raw_input("Δώσε την επιθυμητή επωνυμία για το νέο κατάστημα:")
29
30 # 4.2
31 result=check_name(desired_name, TITLES, OWNERS)
32 if result == "FREE":
33     print "ΔΙΑΘΕΣΙΜΗ ΕΠΩΝΥΜΙΑ"
34 else:
35     print "ΔΕΣΜΕΥΜΕΝΗ ΑΠΟ ", result
```

ΘΕΜΑ 2

2.1.

Να αναπτύξετε ισοδύναμο τμήμα προγράμματος με το παρακάτω τμήμα προγράμματος Python, χρησιμοποιώντας την εντολή επανάληψης for αντί της εντολής επανάληψης while.

```
1  apot= 1
2  i=0
3  while i <=20:
4      apot= apot**i
5      print i
6      i=i+2
7  print apot
```

Μονάδες 10

2.2.

Να αντιστοιχίσετε κάθε στοιχείο της στήλης Α του ακόλουθου πίνακα, με το κατάλληλο στοιχείο της στήλης Β.

Στήλη Α	Στήλη Β
1. !=	Α. Συγκριτικός τελεστής
2. total	Β. Αριθμητικός τελεστής
3. %	Γ. Λογικός Τελεστής
4. float	Δ. Ενσωματωμένη συνάρτηση
5. not	Ε. Μη ενσωματωμένη συνάρτηση
	ΣΤ. Όνομα μεταβλητής

Μονάδες 5

2.3

Στο παρακάτω τμήμα προγράμματος σε Python υπάρχουν πέντε (5) κενά, τα οποία έχουν αριθμηθεί και υπογραμμιστεί. Να γράψετε στο γραπτό σας τον αριθμό του κάθε κενού και δίπλα τι πρέπει να συμπληρωθεί, ώστε το τμήμα προγράμματος να υπολογίζει και να εμφανίζει το άθροισμα των αρνητικών ακεραίων από το -20 έως το 0.

```
1  countdown = (1)
2  sum=0
3  while countdown <(2):
4      print countdown,
5      sum = (3)+countdown
6      countdown = countdown+(4)
7  print (5)
```

Μονάδες 10

ΘΕΜΑ 2

2.1

```
1  apot= 1
2  for i in range(0,21,2):
3      apot= apot ** i
4      print x
5  print apot
```

2.2

1. A

2. ΣT

3. B

4. Δ

5. Γ

2.3

1. -20
2. 0
3. sum
4. 1
5. sum

ΘΕΜΑ 2

2.1 Δίνεται το παρακάτω τμήμα προγράμματος Python

```
1 rows = 4
2 for j in range(1, rows+1):
3     print("* " * j)
4
5 for j in range(rows,0,-1):
6     print("* " * j)
```

A. Τι ακριβώς θα εμφανισθεί μετά την ολοκλήρωση της εκτέλεσης του;

Μονάδες 10

B. Πόσες φορές θα εκτελεστεί η εντολή 3 `print("* " * j);`

Μονάδες 5

2.2. Από τα ακόλουθα δεκαπέντε (15) ονόματα μεταβλητών να επιλέξετε, γράφοντας στο γραπτό σας το αντίστοιχο γράμμα, όσα θεωρούνται έγκυρα στη γλώσσα προγραμματισμού Python:

A. round 1	B. rou_ND	Γ. 2round
Δ. floated	E. ROUND	ΣΤ. round1_
Z. integer	H. break	Θ. python
I. r_ou_nd	K. round%	Λ. _Round1
M. python1	N. ifelse	Ξ. 1Round1

Μονάδες 10

ΘΕΜΑ 2

2.1

A.

*

* *

* * *

* * * *

* * * *

* * *

* *

*

B. 8 φορές

2.2 Τα έγκυρα ονόματα μεταβλητών είναι τα εξής:

1. **B**
2. **Δ**
3. **E**
4. **ΣΤ**
5. **Z**
6. **Θ**
7. **I**
8. **Λ**
9. **M**
10. **N**

ΘΕΜΑ 2

2.1.

Να αναπτύξετε ισοδύναμο τμήμα προγράμματος με το παρακάτω τμήμα προγράμματος Python, χρησιμοποιώντας την εντολή επανάληψης while αντί της εντολής επανάληψης for.

```
1  s=10
2  for x in range (0,100,10):
3      s=s+x
4  print s
```

Μονάδες 10

2.2

Στο παρακάτω τμήμα προγράμματος σε Python υπάρχουν πέντε (5) κενά, τα οποία έχουν αριθμηθεί και υπογραμμιστεί. Να γράψετε στο γραπτό σας τον αριθμό του κάθε κενού και δίπλα τι πρέπει να συμπληρωθεί, ώστε το τμήμα προγράμματος να υπολογίζει το άθροισμα ενός άγνωστου πλήθους εισαγόμενων αριθμών καθώς και να μετράει το πλήθος των μονών από αυτούς μέχρι να δοθεί -1.

```
1  sum =0
2  cnt_odd=0
3  n = int(input('Δώστε έναν θετικό αριθμό (-1 για τέλος):'))
4  while n (1)=-1:
5      sum = sum + (2)
6      if n%2 (3)=0:
7          cnt_odd=(4)+1
8      n = int(input('Δώστε έναν θετικό αριθμό (-1 για τέλος):'))
9  print 'Αθροισμα =', sum
10 print 'Πλήθος μονών =', (5)
```

Μονάδες 15

ΘΕΜΑ 2

2.1

```
1  s=10
2  x=0
3  while x<100:
4      s=s+x
5      x=x+10
6  print s
```

2.2

```
1.  !
2.  n
3.  !
4.  cnt_odd
5.  cnt_odd
```


ΘΕΜΑ 2

2.1.

A. Να αναπτύξετε ισοδύναμο τμήμα προγράμματος με το παρακάτω τμήμα προγράμματος Python, χρησιμοποιώντας την εντολή επανάληψης while αντί της εντολής επανάληψης for.

```
1  apot= 1
2  for i in range(0,21,2):
3      apot= apot ** i
4      print i
5  print apot
```

Μονάδες 10

2.2.

Να αντιστοιχίσετε κάθε στοιχείο της στήλης A του ακόλουθου πίνακα, με το κατάλληλο στοιχείο της στήλης B. Ένα από τα στοιχεία της στήλης B περισσεύει.

Στήλη A	Στήλη B
1. %	A. Συγκριτικός τελεστής
2. sum	B. Αριθμητικός τελεστής
3. ==	Γ. Λογικός Τελεστής
4. raw_input	Δ. Ενσωματωμένη συνάρτηση
5. or	Ε. Μη ενσωματωμένη συνάρτηση
	ΣΤ. Όνομα μεταβλητής

Μονάδες 5

2.3

Στο παρακάτω τμήμα προγράμματος σε Python υπάρχουν πέντε (5) κενά, τα οποία έχουν αριθμηθεί και υπογραμμιστεί. Να γράψετε στο γραπτό σας τον αριθμό του κάθε κενού και δίπλα τι πρέπει να συμπληρωθεί, ώστε το τμήμα προγράμματος να υπολογίζει και να εμφανίζει το άθροισμα των αρνητικών άρτιων ακεραίων από το -20 έως το 0.

```
1  countdown = (1)
2  sum=0
3  while (2)<0:
4      print countdown,
5      sum = (3) +(4)
6      countdown = countdown+(5)
7  print sum
```

Μονάδες 10

ΘΕΜΑ 2

2.1

A.

```
1  apot= 1
2  i=0
3  while i <21:
4      apot= apot**i
5      print i
6      i=i+2
7  print apot
```

2.2

1. B

2. ΣT

3. A

4. Δ

5. Γ

2.3

1. -20
2. countdown
3. sum
4. countdown
5. 2

ΘΕΜΑ 4

Κάθε χρόνο, οι φορολογούμενοι πολίτες υποχρεούνται να δηλώσουν δαπάνες, των οποίων το 30% πρέπει να γίνεται με ηλεκτρονικό τρόπο (κάρτα ή E-Banking). Να αναπτυχθεί πρόγραμμα σε γλώσσα προγραμματισμού Python όπου:

4.1 Να εισάγεται το εισόδημά του φορολογούμενου και η συνολική αξία δαπανών του με ηλεκτρονική συναλλαγή ή κάρτα (να γίνεται έλεγχος εγκυρότητας και τα δύο ποσά για αποτροπή εισαγωγής αρνητικών τιμών).

Μονάδες 4

4.2 Να υπολογίζει και να εμφανίζει το απαιτούμενο ποσό ηλεκτρονικών δαπανών που αντιστοιχεί στο 30% του δηλωθέντος εισοδήματος. Επίσης να υπολογίζει και να εμφανίζει τη διαφορά μεταξύ υποχρεωτικών δαπανών και δαπανών με ηλεκτρονική συναλλαγή ή κάρτα.

Μονάδες 10

4.3 Με χρήση μιας συνάρτησης, που θα δημιουργηθεί, με όνομα `sur_tax` θα μεταβιβάζεται η διαφορά για να υπολογισθεί και να επιστραφεί, στο κύριο πρόγραμμα, η τυχόν κράτηση φόρου. Συγκεκριμένα, αν η διαφορά είναι θετικός αριθμός τότε θα υπολογίζεται, ως φόρος το 22% αυτής της διαφοράς διαφορετικά ο φόρος θα είναι 0.

Μονάδες 8

4.4. Να εμφανίζει το ποσό του φόρου που δεν είναι μηδενικό διαφορετικά να εμφανίζει το μήνυμα «Δεν προκύπτει επιπλέον φόρος».

Μονάδες 3

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1 def sup_tax(diaf): #υπολογισμός προσαύξησης φόρου
2     if diaf > 0:
3         sup_tax= diaf*.22
4     else:
5         sup_tax =0
6     return sup_tax
7
8 # Κύριο πρόγραμμα
9 # 4.1 εισαγωγή εισοδήματος και αξίας ηλεκτρονικών συναλλαγών
10 income = float(input("Πληκτρολόγησε το εισόδημα σου: "))
11 while (income<0):      # 4.1 έλεγχος εγκυρότητας εισαγόμενου αριθμού
12     print 'Αποδεκτές τιμές: θετικοί ακέραιοι '
13     income=float(input('Ξαναπληκτρολόγησε το εισόδημα σου: ')) #4.1 επανεισαγωγή εισοδήματος
14 expenses = float(input("Πληκτρολόγησε την αξία των ηλεκτρονικων δαπανών σου: "))
15 while (expenses<0):    # 4.1 έλεγχος εγκυρότητας εισαγόμενου αριθμού
16     print 'Αποδεκτές τιμές: θετικοί ακέραιοι '
17     expenses=float(input('Ξαναπληκτρολόγησε την αξία των ηλεκτρονικων δαπανών σου: '))
18
19 apait_expenses = income *.3 #4.2 υπολογισμός απαιτούμενου ποσού ηλεκτρονικών δαπανών
20 print 'Απαιτούμενο ποσό ηλεκτρονικών δαπανών: ',apait_expenses
21 diafora= apait_expenses-expenses
22 print 'Διαφορά δαπανών: ',diafora
23
24 supp_tax= sup_tax(diafora) #4.3 κλήση συνάρτησης
25
26 if supp_tax > 0:  #4.4.εμφάνιση προσαύξησης φόρου
27     print 'Προκύπτει επιπλέον φόρος ο οποίος είναι: ', supp_tax
28 else:
29     print 'Δεν προκύπτει επιπλέον φόρος'
```

ΘΕΜΑ 4

Μια κατασκευαστική εταιρεία πουλάει οροφδιαμερίσματα (ένα διαμέρισμα ανά όροφο) σε μια οικοδομή έχει 9 ορόφους (το ισόγειο θεωρείται ότι είναι ο πρώτος όροφος και έχει αριθμό ορόφου 0). Από το ισόγειο μέχρι και τον 6^ο όροφο το εμβαδόν κάθε διαμερίσματος είναι 160 τ.μ., στον 7^ο είναι 140 τ.μ. και στον 8ο 120 τ.μ. Η τιμή πώλησης ανά τ.μ είναι 1000€ και προστίθεται επιπλέον 3% ανά όροφο. Να αναπτύξετε πρόγραμμα σε γλώσσα προγραμματισμού Python το οποίο:

4.1 Να εισάγει τα τετραγωνικά μέτρα των 9 διαμερισμάτων σε μία λίστα (apart)

Μονάδες 2

4.2 Να διαβάζει επανειλημμένα τον αριθμό ορόφου που θέλει να αγοράσει ένας υποψήφιος αγοραστής. Να γίνεται έλεγχος ορθότητας ώστε να εισάγονται μόνο αποδεκτές τιμές από το 0 μέχρι και το 8. Η εκτέλεση του προγράμματος θα διακόπτεται όταν δοθεί αρνητικός αριθμός για όροφο.

Μονάδες 7

4.3 Με χρήση συνάρτησης που θα δημιουργηθεί και θα έχει όνομα `total_value` να υπολογίζεται και να επιστρέφεται, στο κύριο πρόγραμμα, η αρχική αξία του διαμερίσματος με βάση τον αριθμό ορόφου που θα δίνεται.

Μονάδες 10

4.4. Στο ερώτημα «Θα πληρώσετε όλο το ποσό; (ΝΑΙ/ΟΧΙ)» αν η απάντηση είναι θετική τότε να υπολογίζει έκπτωση 15% επί της αρχικής αξίας και να εμφανίζει την τελική αξία, μετά την έκπτωση, με το σχετικό μήνυμα «Μία πληρωμή: », XXXXXXX € αλλιώς να υπολογίζει την αξία της κάθε δόσης για 12 μηνιαίες ισόποσες δόσεις επί της αρχικής αξίας και να εμφανίζει το μήνυμα «Η μηνιαία δόση: » XXXXX€

Μονάδες 6

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1  def total_value (level): #4.2 Συνάρτηση υπολογισμού αρχικής τιμής διαμερίσματος
2      value = apart[level]*1000
3      value = value + (value*level*0.03)
4
5      return value
6
7  #Κύριο πρόγραμμα
8  apart =[160,160,160,160,160,160,160,140,120] #4.1 εισαγωγή τ.μ. 9 διαμερισμάτων
9
10 orofos=int(input('Δώσε όροφο από 0 έως και 8, διακοπή με -1 : ')) #4.2 εισαγωγή ορόφου
11 while (orofos<-1) or (orofos>8):      # 4.2 έλεγχος εγκυρότητας εισαγόμενου αριθμού
12     print 'Αποδεκτή τιμή μεταξύ 0-8'
13     orofos=int(input('Δώσε όροφο από 0 έως και 8, διακοπή με -1 : ')) #4.2 εισαγωγή ορόφου
14
15 while orofos !=-1:
16     timi = total_value(orofos) # 4.3 υπολογισμός αρχικής αξίας διαμερίσματος
17     apant=raw_input('Θα πληρώσετε όλο το ποσό (ΝΑΙ/ΟΧΙ); ')
18     if apant=='ΝΑΙ':
19         timi=timi-(timi*15/100) # 4.4 υπολογισμός έκπτωσης 15% επί της αρχικής
20         print 'Μία πληρωμή: ', timi, 'ευρώ'
21     else:
22         dosi= float(timi/12) # 4.4 υπολογισμός 12 δόσεων
23         print 'Η μηνιαία δόση: ', dosi, 'ευρώ'
24     print '\n'
25
26     orofos=int(input('Δώσε όροφο από 0 έως και 8, διακοπή με -1 : ')) #4.2 εισαγωγή ορόφου
27     while (orofos<-1) or (orofos>8):      # 4.2 έλεγχος εγκυρότητας εισαγόμενου αριθμού
28         print 'Αποδεκτή τιμή μεταξύ 0-8'
29         orofos=int(input('Δώσε όροφο από 0 έως και 8, διακοπή με -1 : ')) #4.2 εισαγωγή ορόφου
```


ΘΕΜΑ 4

Δίσεκτο ονομάζεται το έτος το οποίο έχει μια ημέρα παραπάνω το μήνα Φεβρουάριο, δηλαδή 29 αντί για 28 ημέρες. Αυτό συμβαίνει κάθε τέσσερα χρόνια υπό κάποιες προϋποθέσεις (βλέπε συμπληρωματικές οδηγίες). Να γραφεί πρόγραμμα σε γλώσσα προγραμματισμού Python το οποίο:

4.1 Να διαβάζει το αρχικό και το τελικό έτος μιας περιόδου. Το πρόγραμμα θα πρέπει να αποτρέπει είσοδο αρνητικών τιμών και μεγαλύτερων του 3000.

Μονάδες 5

4.2 Σε περίπτωση που το αρχικό είναι μεγαλύτερο του τελικού έτους να γίνεται αντιμετάθεση.

Μονάδες 5

4.3 Να εντοπίζει όλα τα δίσεκτα έτη της επιλεγείσης χρονικής περιόδου, από το αρχικό προς το τελικό έτος, και να εμφανίζει σχετικό μήνυμα μόνο για κάθε τέτοιο έτος π.χ. «Το 2024 είναι δίσεκτο», «Το 2028 είναι δίσεκτο» κ.ο.κ

Μονάδες 10

4.4 Να υπολογίζει και να εμφανίζει το πλήθος των δίσεκτων της επιλεγείσης χρονικής περιόδου.

Μονάδες 5

Συμπληρωματικές οδηγίες:

Κάποιο έτος είναι δίσεκτο όταν το υπόλοιπο της ακεραίας διαίρεσης του έτους με το 4 είναι μηδέν και ταυτόχρονα το υπόλοιπο της ακεραίας διαίρεσης του έτους με το 100 είναι διαφορετικό του μηδενός ή το υπόλοιπο της ακεραίας διαίρεσης του έτους με το 400 είναι μηδέν.

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1  start_year=int(input('Δώσε αρχικό έτος περιόδου: '))    # 4.1 εισαγωγή αρχικού έτους
2  while (start_year<=0) or (start_year>=3000):          # 4.1 έλεγχος εγκυρότητας αρχικού έτους
3      print 'Τιμές εκτός ορίων. Επιτρεπτά όριο 1-2999'
4      start_year=int(input('Ξαναδώσε αρχικό έτος περιόδου: '))
5
6  end_year=int(input('Δώσε τελικό έτος περιόδου: '))    # 4.1 εισαγωγή τελικού έτους
8  while (end_year<=0) or (end_year>=3000):              # 4.1 έλεγχος εγκυρότητας τελικού έτους
9      print 'Τιμές εκτός ορίων. Επιτρεπτά όριο 1-2999'
10     end_year=int(input('Ξαναδώσε τελικό έτος περιόδου: '))
12
13  if start_year > end_year: # αντιμετάθεση όταν αρχικό μεγαλύτερο του τελικού έτους
14      temp= start_year
15      start_year = end_year
16      end_year = temp
17
18  metr=0
19  for year in range (start_year, end_year+1):
20      if (year%4==0) and (year%100!=0) or (year%400==0):
21          print 'To ', year, 'είναι δίσεκτο'
22          metr= metr+1
23
24  print 'Πλήθος δίσεκτων περιόδου ', metr
```

ΘΕΜΑ 4

Κάθε μέρα του Αυγούστου προσεγγίζουν το λιμάνι ενός ελληνικού νησιού τέσσερα κρουαζιερόπλοια και αποβιβάζουν επιβάτες για να επισκεφθούν το δημοφιλές μουσείο αρχαιοτήτων. Το τοπικό λιμεναρχείο καταγράφει μερικά στοιχεία για στατιστικούς λόγους.

Να αναπτυχθεί πρόγραμμα σε γλώσσα προγραμματισμού Python όπου:

4.1. Να διαβάζονται, το όνομα του κρουαζιερόπλοιου, το πλήθος ανδρών επισκεπτών, το πλήθος γυναικών επισκεπτών και το πλήθος παιδιών ανεξαρτήτως φύλου για όλα τα κρουαζιερόπλοια. Τα στοιχεία αυτά να καταχωρούνται στις λίστες NAME, MALE, FEMALE και CHILD αντίστοιχα. Τα καταχωρούμενα στοιχεία θεωρούνται έγκυρα.

Μονάδες 4

4.2. Να υπολογίζεται και να εμφανίζεται, με κατάλληλο μήνυμα, το συνολικό άθροισμα ανδρών επισκεπτών, το συνολικό άθροισμα γυναικών επισκεπτών και το αντίστοιχο των παιδιών από όλα τα κρουαζιερόπλοια.

Μονάδες 7

4.3. Να εντοπίζεται και να εμφανίζεται το όνομα και το πλήθος παιδιών του κρουαζιερόπλοιου με τα περισσότερα παιδιά.

Μονάδες 8

4.4. Να υπολογίζεται και να εμφανίζεται το σύνολο του εισπραχθέντος παράβολου επίσκεψης που πληρώνουν μόνο οι άνδρες και γυναίκες επιβάτες. Το παράβολο ανέρχεται σε 0,80€ ανά επιβάτη.

Μονάδες 6

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές.

Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
    print b
```

ΘΕΜΑ 4

```
1  NAME=[] #δημιουργία κενής λίστας ονομασίας σκαφών
2  MALE=[] #δημιουργία κενής λίστας πλήθους ανδρών επισκεπτών
3  FEMALE=[] #δημιουργία κενής λίστας γυναικών επισκεπτών
4  CHILD=[] #δημιουργία κενής παιδιών επισκεπτών
5
6  ship_nr=4 # πλήθος σκαφών
8  tot_men=0 ; tot_women=0 ; tot_children=0 # αθροιστές επισκεπτών
9  max_child=-1 # μετρητής παιδιών
10 paravolo=0
12 for i in range(ship_nr):
13     ship_name=raw_input('Δώσε ονομασία σκάφους: ') #4.1 εισαγωγή ονομασία σκάφους
14     NAME.append(ship_name)
15
16     male_count=int(input('Δώσε πλήθος ανδρών : ')) #4.1 εισαγωγή πλήθους ανδρών
17     MALE.append(male_count)
18     female_count=int(input('Δώσε πλήθος γυναικών : ')) #4.1 εισαγωγή πλήθους γυναικών
19     FEMALE.append(female_count)
20     child_count=int(input('Δώσε πλήθος παιδιών : ')) #4.1 εισαγωγή πλήθους παιδιών
21     CHILD.append(child_count)
22
23     tot_men= tot_men+male_count #4.2 συνολικό άθροισμα ανδρών επισκεπτών,
24     tot_women=tot_women+female_count #4.2 συνολικό άθροισμα γυναικών επισκεπτών
25     tot_children=tot_children+child_count #4.2 συνολικό άθροισμα παιδιών επισκεπτών
26
27 #4.3. εντοπισμός ονόματος και πλήθους παιδιών του κρουαζιερόπλοιου με τα περισσότερα παιδιά
28     if child_count > max_child:
29         max_child= child_count
30         max_name = NAME[i]
31
32 #4.3. εμφάνιση ονόματος και πλήθους του κρουαζιερόπλοιου με τα περισσότερα παιδιά
33     print 'Το σκάφος με όνομα: ', max_name, ' είχε τα περισσότερα παιδιά, δηλαδή: ', max_child
34
35     paravolo = (tot_men+tot_women)*0.8 #4.4 υπολογισμός παράβολου επίσκεψης ανδρών & γυναικών
36     print 'Το συνολικό εισπραχθέν παράβολο ανέρχεται σε: ', paravolo, ' ευρώ'
```

ΘΕΜΑ 4

Να αναπτυχθεί πρόγραμμα σε γλώσσα προγραμματισμού Python με το οποίο:

4.1. Να δημιουργηθεί μια συνάρτηση με όνομα `give_numbers`, η οποία να δημιουργεί μια λίστα με όνομα `lista50` στην οποία θα εισάγονται από το πληκτρολόγιο 50 τυχαίοι ακέραιοι αριθμοί. Οι 50 αριθμοί πρέπει να είναι από το 1 μέχρι και το 20. Να γίνεται έλεγχος ορθότητας ώστε να εισάγονται μόνο αποδεκτές τιμές από το 1 μέχρι και το 20.

Μονάδες 10

4.2. Να εντοπίζεται, στο κύριο πρόγραμμα, η τιμή της θέσης 25 της λίστας `lista50` και να εμφανίζεται με κατάλληλο μήνυμα.

Μονάδες 5

4.3. Με χρήση συνάρτησης που επίσης θα δημιουργηθεί και θα έχει όνομα `count25`, να υπολογίζεται το πλήθος εμφάνισης της τιμής της θέσης 25 της λίστας `lista50` και να επιστρέφεται στο κύριο πρόγραμμα όπου και θα εμφανίζεται με κατάλληλο μήνυμα.

Μονάδες 10

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές.

Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1  def give_numbers(lista50):
2      for i in range(50): # 4.1 εισαγωγή 50 ακεραίων στη λίστα
3          akeraios=int(input('Δώσε ένα ακέραιο από 1 έως και 20: '))#4.1 εισαγωγή ακέραιου
4          while (akeraios<1) or (akeraios>20):      # 4.1 έλεγχος εγκυρότητας ακέραιου
5              print 'Αποδεκτή τιμή μεταξύ 1-20'
6              akeraios=int(input('Δώσε ένα ακέραιο από 1 έως και 20: ')) #4.2 επαναεισαγωγή
7          lista50.append(akeraios)
8      return lista50 #4.1 επιστροφή λίστας
9
10 def count25(lista50,plithos, nr25): # 4.3 συνάρτηση εύρεσης πλήθους
11     metr = 0
12     # 4.3 καταμέτρηση τιμών όμοιων με την τιμή της θέσης 25
13     for i in range(plithos):
14         if lista50[i] == nr25:
15             metr = metr+1 #4.3 μετρητής πλήθους
16     return metr
17
18 # Κύριο πρόγραμμα
19 lista50 = []      # 4.1 Δήλωση λίστας
20 give_numbers(lista50) #4.1 κλήση συνάρτησης εισαγωγής με 50 ακέραιους
21
22 thesi25 = lista50[25] # 4.2 εύρεση τιμής της θέσης 25
23 print 'Ο 25ος αριθμός είναι: ', thesi25
24
25 pl25 = count25(lista50,50, thesi25) # 4.3 κλήση συνάρτησης εύρεσης πλήθους
26 print 'Πλήθος εμφάνισης :', pl25
```

ΘΕΜΑ 2

2.1.

A. Να αναπτύξετε ισοδύναμο τμήμα προγράμματος με το παρακάτω τμήμα προγράμματος Python, χρησιμοποιώντας την εντολή επανάληψης for αντί της εντολής επανάληψης while.

```
1 s=10
2 x=0
3 while x<100:
4     s=s+x
5     x=x+10
6 print s
```

Μονάδες 10

2.2.

Στο παρακάτω τμήμα προγράμματος σε Python υπάρχουν πέντε (5) κενά, τα οποία έχουν αριθμηθεί και υπογραμμιστεί. Να γράψετε στο γραπτό σας τον αριθμό του κάθε κενού και δίπλα τι πρέπει να συμπληρωθεί, ώστε το τμήμα προγράμματος να υπολογίζει και να εμφανίζει το γινόμενο των αριθμών από 1 μέχρι 9.

```
1 prod=(1)
2 for i in range(2,(2)):
3     prod = prod*(3)
4 print 'το γινόμενο είναι: '(4) (5)
```

Μονάδες 10

2.3. Να συμπληρώσετε στο γραπτό σας την στήλη «**Αποτέλεσμα στην οθόνη**» όπου θα γράφετε το γράμμα **α, β, γ, δ** και **ε** και μετά το αποτέλεσμα που θα εμφανιστεί στο περιβάλλον της γλώσσας Python μετά την εκτέλεση της/των εντολής/εντολών της αντίστοιχης στήλης «**Εντολές**».

Εντολές	Αποτέλεσμα στην Οθόνη
α. a,b,c,d= 'a', 'be','ba',' ' print b, d, c, a	α.
β. say='ole ' print say*3	β.
γ. x = 5 y = 10 x = x * y y = x / y x = x / y print x, y	γ.
δ. first=8 last=2 last=first/last first=last%first print last, first	δ.
ε. a = 5 b = 4 if a >= b: a=a**(b%2) else: a=a%2 print a	ε.

Μονάδες 5

ΘΕΜΑ 2

2.1

```
1 s=10
2 for x in range (0,100,10):
3     s=s+x
4 print s
```

2.2

1. 1
2. 10
3. i
4. ,
5. prod

2.3

Αποτέλεσμα στην Οθόνη

α. be {κενό} ba a

β. ole ole ole

γ. 10 5

δ. 4 4

ε. 1

ΘΕΜΑ 2

2.1 Να συμπληρώσετε στο γραπτό σας τα αριθμημένα κενά με τις λέξεις που λείπουν επιλέγοντας από αυτές που δίνονται στο τέλος των προτάσεων:

Το πρώτο στάδιο για την επίλυση ενός προβλήματος είναι η _____ (1)_____ .

Με τον όρο _____ (2)_____, εννοούμε τα επιμέρους _____ (3)_____ που αποτελούν το πρόβλημα καθώς και τον τρόπο με τον οποίο αυτά συνδέονται και αλληλεπιδρούν.

Η _____ (4)_____ αναφέρεται κυρίως στην ταχύτητα εκτέλεσης του αλγόριθμου και στην ποσότητα της κύριας μνήμης που χρησιμοποιεί.

_____ (5)_____ χαρακτηρίζεται το πρόβλημα το οποίο μπορεί να επιλυθεί με την βοήθεια ενός υπολογιστή.

Λέξεις προς επιλογή:

A. Κατανόηση **B.** Αφαίρεση **Γ.** Δομή **Δ.** Ανάλυση **Ε.** Στοιχεία (τμήματα) **Στ.** Μαθηματικό **Ζ.** Αποδοτικότητα **Η.** Υπολογιστικό

Μονάδες 10

2.2 Δίνεται το παρακάτω πρόγραμμα σε Python, το οποίο δέχεται έναν αριθμό από έναν χρήστη και υπολογίζει και εμφανίζει στην οθόνη, το άθροισμα όλων των αριθμών από το 1 έως και τον δοσμένο αυτόν αριθμό. Το πρόγραμμα έχει σφάλματα.

```
1 s = 1
2 n = int(input("Εισάγετε αριθμό: "))
3 for i in range(1, n)
4     s =+ i
5 print \n
6 print "Sum is: ", n
```

Να εντοπίσετε τα σφάλματα αυτά και να συμπληρώσετε για κάθε σφάλμα μια ξεχωριστή γραμμή στον πίνακα που ακολουθεί:

Αριθμός γραμμής	Σφάλμα	Διόρθωση σφάλματος	Κατηγορία σφάλματος
-----------------	--------	--------------------	---------------------

--	--	--	--

Μονάδες 15

ΘΕΜΑ 2

2.1

1 – Α(Κατανόηση)

2 – Γ(Δομή)

3 – Ε (Στοιχεία (τμήματα))

4 – Ζ (Αποδοτικότητα)

5 – Η(Υπολογιστικό)

2.2

Αριθμός γραμμής	Σφάλμα	Διόρθωση σφάλματος	Κατηγορία σφάλματος
1	<code>s = 1</code>	<code>s=0</code>	Σημασιολογικό ή λογικό σφάλμα
3	<code>for i in (1, n)</code>	<code>for i in (1, n+1)</code>	Σημασιολογικό ή λογικό σφάλμα
4	<code>s =+i</code>	<code>s +=i</code>	Συντακτικό λάθος
5	<code>print \n</code>	<code>print "\ n"</code>	Συντακτικό λάθος
6	<code>print "Sum is: ",n</code>	<code>print "Sum is: ",s</code>	Σημασιολογικό ή λογικό σφάλμα

ΘΕΜΑ 4

Στο Πανελλήνιο μαθητικό πρωτάθλημα στίβου, στο αγώνισμα του άλματος σε ύψος, συμμετέχουν στους προκριματικούς 20 μαθητές από όλα τα σχολεία της Επικράτειας. Στον τελικό του αγωνίσματος περνούν όσοι μαθητές σημειώσουν επίδοση μεγαλύτερη ή ίση από 1.2 μέτρα. Κάθε μαθητής έχει δικαίωμα να κάνει 3 προσπάθειες. Να αναπτύξετε πρόγραμμα σε Python, το οποίο:

4.1 Να διαβάζει το όνομα και τις επιδόσεις (θετικός αριθμός για έγκυρο άλμα και μηδέν για άκυρο) των αλμάτων κάθε μαθητή και να υπολογίζει την καλύτερη επίδοσή του. Απαιτείται έλεγχος εγκυρότητας δεδομένων για κάθε επίδοση.

Μονάδες 10

4.2 Να ελέγχει εμφανίζοντας κατάλληλο μήνυμα στην οθόνη, αν ο μαθητής προκρίθηκε ή όχι στον τελικό.

Μονάδες 5

4.3 Να εμφανίζει στην οθόνη το μήνυμα «Δεν προκρίθηκε κανένας μαθητής» στην περίπτωση που δεν προκρίθηκε κανένας μαθητής. Διαφορετικά, να εμφανίζει στην οθόνη πόσοι μαθητές προκρίθηκαν και ποια ήταν η καλύτερη επίδοση μαζί με το όνομα του μαθητή που τη σημείωσε. (Να θεωρήσετε ότι δεν υπάρχουν δύο ή περισσότεροι μαθητές με την ίδια καλύτερη επίδοση).

Μονάδες 10

Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
    print b
```

ΘΕΜΑ 4

```
1  # Αρχικοποίηση μεταβλητών
2  met=0
3  maxtotal=0
4  # 4.1
5  for i in range(1,21):
6      name=raw_input("Εισάγετε όνομα μαθητή: ")
7      max=-1
8      for k in range (1, 4):
9          ep=float(input("Δώστε τα μέτρα της προσπάθειας ή μηδέν (0) για άκυρη προσπάθεια: " ))
10     # Έλεγχος εγκυρότητας δεδομένων
11         while(ep<0):
12             ep=float(input("Δώστε τα μέτρα της προσπάθειας ή μηδέν (0) για άκυρη προσπάθεια: "))
13     # Υπολογισμός καλύτερης επίδοσης για κάθε μαθητή
14         if ep>=1.2 and ep>max:
15             max=ep
16     # 4.2
17         if max>=1.2:
18             print "Ο μαθητής", name, "προκρίθηκε στον τελικό!"
19             met+=1
20         else:
21             print "Ο μαθητής", name, "ΔΕΝ προκρίθηκε στον τελικό!"
22     # Υπολογισμός καλύτερης επίδοσης μαζί με όνομα του μαθητή
23         if maxtotal<max:
24             maxtotal=max
25             nametotal=name
26     # 4.3
27     if met>0:
28         print "Προκρίθηκαν στον τελικό", met, "μαθητές"
29         print "Ο μαθητής", nametotal, "έκανε την καλύτερη επίδοση", maxtotal
30     else:
31         print "Δεν προκρίθηκε κανένας μαθητής"
```

ΘΕΜΑ 2

2.1 Να συμπληρώσετε τα στοιχεία της στήλης **B** με τα αποτελέσματα που παράγονται κατά την εκτέλεση των λογικών εκφράσεων της στήλης **A**

Στήλη A – Λογικές εκφράσεις	Στήλη B - Αποτέλεσμα
1. $((A > B) \text{ and } (C < A)) \text{ or } (C > 5)$ όταν $A = -3, B = 7, C = 5$	
2. $\text{len}(\text{"Papadopoulos"}) > 10 \text{ and } \text{len}(\text{"Papadakis"}) > 10$	
3. $\text{not}(A + B < 10)$ όταν $A = 5, B = 7$	
4. $(\text{year} \% 4 == 0) \text{ and } (\text{year} \% 100 != 0) \text{ or } (\text{year} \% 400 == 0)$ όταν $\text{year} = 2018$	
5. $\text{not}(5 == 5 \text{ or } (1 != 0 \text{ and } 6 != 7))$	

Μονάδες 10

2.2 Το μοντέλο του καταρράκτη υποδιαίρει τη διαδικασία ανάπτυξης ενός συστήματος λογισμικού στις ακόλουθες φάσεις, οι οποίες εμφανίζονται αριθμημένες, αλλά όχι με τη σωστή σειρά:

1. Σχεδίαση
2. Ανάλυση απαιτήσεων
3. Λειτουργία και συντήρηση
4. Ολοκλήρωση
5. Υλοποίηση

Να γράψετε στο γραπτό σας τη σωστή σειρά των φάσεων.

Μονάδες 5

2.3 Να συμπληρώσετε στο γραπτό σας τα αριθμημένα κενά από το (1) μέχρι και το (10) στον παρακάτω κώδικα σε Python, ο οποίος διαβάζει 20 θετικούς ακераίους αριθμούς και καταχωρεί τους άρτιους σε μια λίστα $a[]$ και τους περιττούς σε μια λίστα $p[]$. Στη συνέχεια, εμφανίζει στην οθόνη το πλήθος των άρτιων και το πλήθος των περιττών αριθμών, καθώς και υπολογίζει και εμφανίζει στην οθόνη το μέσο όρο των άρτιων αριθμών. Να θεωρήσετε ότι έχει δοθεί τουλάχιστον ένας άρτιος.


```
1 a= __ (1) __
2 p= __ (2) __
3 s = 0
4 for i in range( __ (3) __):
5     num=int(input("Εισάγετε αριθμό: "))
6     if num __ (4) __ 0:
7         __ (5) __
8     else:
9         __ (6) __
10 for n in a:
11     s= __ (7) __
12 mo= __ (8) __
13 print "Πλήθος άρτιων: ", __ (9) __ , "Πλήθος περιττών: ", __ (10) __
14 print "Ο μέσος όρος των άρτιων αριθμών είναι: ", mo
```

Μονάδες 10

QEMA 2

2.1

1.False

2.False

3.True

4.False

5.False

2.2

$2 - 1 - 5 - 4 - 3$

2.3

(1) []

(2) []

(3) 20

(4) %2==

(5) a.append(num)

(6) p.append(num)

(7) s+n

(8) s/len(a)

(9) len(a)

(10) len(p)

ΘΕΜΑ 4

Στις ενδοσχολικές εξετάσεις των σχολείων του Ιουνίου επιτρέπεται στους μαθητές η χρήση απλής αριθμομηχανής (κομπιουτεράκι) για την εκτέλεση απλών μαθηματικών υπολογισμών. Να αναπτύξετε πρόγραμμα σε Python το οποίο, υλοποιεί τις παρακάτω λειτουργίες σε μια τέτοια αριθμομηχανή:

4.1

A. Να δημιουργεί τη συνάρτηση `menu()` η οποία θα εμφανίζει στην οθόνη το παρακάτω μενού επιλογών:

1. Πρόσθεση
2. Αφαίρεση
3. Πολλαπλασιασμός
4. Διαίρεση
5. Ύψωση στο τετράγωνο
6. Έξοδος

Μονάδες 5

B. Στη συνέχεια, η συνάρτηση ζητά από τον μαθητή, με κατάλληλο μήνυμα, να πληκτρολογήσει μια επιλογή από το 1 μέχρι και το 6, την οποία επιστρέφει στο κύριο πρόγραμμα. Απαιτείται να γίνεται έλεγχος εγκυρότητας δεδομένων για τους αριθμούς που εισάγονται.

Μονάδες 5

4.2 Όταν επιλέγονται τα νούμερα από 1 μέχρι και 5, να εισάγονται από τον μαθητή δύο πραγματικοί αριθμοί και στη συνέχεια να εκτελείται η αντίστοιχη πράξη για τους παραπάνω αριθμούς, σύμφωνα με το μενού του ερωτήματος **4.1** εμφανίζοντας στην οθόνη το αποτέλεσμα της.

Μονάδες 10

4.3 Όταν επιλεγεί το νούμερο 6, το πρόγραμμα εμφανίζει στην οθόνη το μήνυμα «Τερματισμός», διαφορετικά εμφανίζεται το μενού επιλογών και επαναλαμβάνεται η διαδικασία της επιλογής για μια νέα πράξη με δύο νέους αριθμούς.

Μονάδες 5

Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1  print "Αριθμομηχανή"
2  # 4.1.A Δημιουργία συνάρτησης
3  def menu():
4      print "1. Πρόσθεση"
5      print "2. Αφαίρεση"
6      print "3. Πολλαπλασιασμός"
7      print "4. Διαίρεση"
8      print "5. Ύψωση στο τετράγωνο"
9      print "6. Έξοδος"
10     print "\n"
11 # 4.1.B Έλεγχος εγκυρότητας δεδομένων
12     c=int(input("Παρακαλώ διάλεξε νούμερο: "))
13     while not(c>=1 and c<=6):
14         c=int(input("Προσοχή! Διάλεξε νούμερο από 1 μέχρι και 6: "))
15     return c
16 choice=menu()
17 # 4.3 Επιλογή από 1 μέχρι 5
18 while (choice!=6):
19     x = float(input("Παρακαλώ πληκτρολόγησε τον πρώτο αριθμό: "))
20     y = float(input("Παρακαλώ πληκτρολόγησε τον δεύτερο αριθμό: "))
21     print "\n"
22 # 4.2 Εκτέλεση πράξεων
23     if choice == 1:
24         print "x + y =", x+y
25     elif choice == 2:
26         print "x - y =", x-y
27     elif choice == 3:
28         print "x * y =", x*y
29     elif choice == 4:
30         if y!=0:
31             print "x / y =", x/y
32         else:
33             print "Δεν μπορεί να γίνει η διαίρεση"
34     elif choice == 5:
35         print "x ^ y =", x**y
36     print "\n"
37     choice=menu()
38 # 4.3
39 else:
40     print "Τερματισμός"
```

ΘΕΜΑ 2

2.1 Για κάθε ένα από τα παρακάτω τμήματα εντολών σε Python της στήλης **A** να συμπληρώσετε το αποτέλεσμα που θα εμφανιστεί στην οθόνη στη στήλη **B**

Στήλη A- Τμήμα εντολών σε Python	Στήλη B – Αποτέλεσμα στην οθόνη
1. <code>var = "Δευτέρα" * 2 * 3</code> <code>print var</code>	
2. <code>v1 = "1"</code> <code>v2 = "2"</code> <code>v3 = "3"</code> <code>print v1 + v2 + v3</code>	
3. <code>x = 0</code> <code>while (x < 10):</code> <code> x+=2</code> <code>print x</code>	
4. <code>names = ["Mary", "George", "Ellen"]</code> <code>names.append("John")</code> <code>print names</code>	
5. <code>for i in range(10, 15, 1):</code> <code> print i, ", " ,</code>	

Μονάδες 10

2.2 Να συμπληρώσετε στο γραπτό σας τα αριθμημένα κενά από το (1) μέχρι και το (5) στον παρακάτω κώδικα σε Python, ο οποίος ζητάει έναν ακέραιο θετικό αριθμό και υπολογίζει και εμφανίζει στην οθόνη το γινόμενο όλων των αριθμών από το 1 μέχρι και τον αριθμό αυτό.

```
1 num = int(input("Εισάγετε τον αριθμό n: "))
2 if num __(1)__:
3     print "Δεν μπορεί να υπολογιστεί το γινόμενο για αρνητικό αριθμό ή 0"
4 else:
5     ginomeno = __(2)__
6     for i in range(__(3)__ , __(4)__):
7         ginomeno = __(5)__
8     print "Το γινόμενο είναι: ", ginomeno
```

Μονάδες 15

ΘΕΜΑ 2

2.1

1 - ΔευτέραΔευτέραΔευτέραΔευτέραΔευτέραΔευτέρα

2 - 123

3 - 10

4 - ['Mary', 'George', 'Ellen', 'John']

5 - 10, 11, 12, 13, 14,

2.2

(1) <=

(2) 1

(3) 1

(4) num+1

(5) ginomeno*i

ΘΕΜΑ 4

Μία εταιρεία μεταφοράς δεμάτων διαθέτει ένα φορτηγό στο οποίο τοποθετούνται όλα τα δέματα προς παράδοση. Ο κάθε παραλήπτης πληρώνει κατά την παραλαβή του δέματος το ποσό που του αναλογεί. Να αναπτύξετε πρόγραμμα σε Python, το οποίο:

4.1 Για κάθε δέμα να δέχεται τον κωδικό του δέματος, ο οποίος αποτελείται από γράμματα και αριθμούς και το ποσό είσπραξης από τον παραλήπτη (σε ευρώ). Η καταχώρηση σταματάει όταν δοθεί ως κωδικός δέματος το '1111'. Να θεωρήσετε ότι το ποσό είναι θετικός αριθμός και δεν απαιτείται έλεγχος δεδομένων.

Μονάδες 5

4.2 Να εμφανίζει στην οθόνη τον συνολικό αριθμό των δεμάτων προς παράδοση και το συνολικό ποσό είσπραξης για όλα τα δέματα.

Μονάδες 10

4.3 Να εμφανίζει στην οθόνη το μεγαλύτερο και το μικρότερο ποσό είσπραξης, καθώς και τους αντίστοιχους κωδικούς δεμάτων. Να θεωρήσετε ότι δεν υπάρχουν δύο ή περισσότερα δέματα με το ίδιο ποσό είσπραξης.

Μονάδες 10

Να θεωρήσετε ότι υπάρχει τουλάχιστον ένα δέμα προς παράδοση.

Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
    print b
```

ΘΕΜΑ 4

```
1  # Αρχικοποίηση μεταβλητών
2  kodikos='0'
3  demata = 0
4  total = 0
5  max = 0
6  min = 999999
7  max_kodikos = kodikos
8  min_kodikos = kodikos
9  # 4.1 Εισαγωγή κωδικών
10 kodikos = raw_input("Εισάγετε κωδικό δέματος: ")
11 print "\n"
12 while kodikos != '1111' :
13     # Πλήθος δεμάτων
14     demata+=1
15     poso = float(input("Ποσό πληρωμής από τον παραλήπτη: "))
16     # Συνολικό ποσό εισπραξης
17     total = total + poso
18     # 4.3 Εύρεση μεγαλύτερου ποσού
19     if poso > max :
20         max = poso
21         max_kodikos = kodikos
22     # Εύρεση μικρότερου ποσού
23     if poso<min:
24         min = poso
25         min_kodikos = kodikos
26     print "\n"
27     kodikos = raw_input("Εισάγετε επόμενο κωδικό δέματος: ")
28     print "\n"
29 # 4.2
30 print "Πλήθος δεμάτων για παράδοση : " , demata , "\n"
31 print "Συνολικό ποσό εισπραξης: " , total, "\n"
32 # 4.3
33 print "Το δέμα με κωδικό " , max_kodikos, " έχει το μεγαλύτερο ποσό εισπραξης: " , max, "\n"
34 print "Το δέμα με κωδικό " , min_kodikos, " έχει το μικρότερο ποσό εισπραξης: " , min
```

ΘΕΜΑ 2

2.1 Να αντιστοιχήσετε κάθε στοιχείο της στήλης **A** με ένα τουλάχιστον στοιχείο της στήλης **B**, λαμβάνοντας υπόψη ότι ένα στοιχείο της στήλης **A** μπορεί να αντιστοιχεί σε περισσότερα από ένα στοιχεία της στήλης **B**

Στήλη A – Είδη σφαλμάτων	Στήλη B - Περιγραφή
1. Συντακτικά λάθη 2. Χρόνου εκτέλεσης 3. Σημασιολογικά ή λογικά σφάλματα	A. αποτελούν προβλήματα σε ένα πρόγραμμα το οποίο τρέχει χωρίς να παράγει κάποιο μήνυμα λάθους, αλλά δεν κάνει αυτό που θα έπρεπε
	B. παράγονται από την Python όταν διερμηνεύει τον πηγαίο κώδικα
	Γ. παράγονται από τον διερμηνευτή, αν πάει κάτι στραβά κατά την εκτέλεση του προγράμματος
	Δ. το αποτέλεσμα από τον υπολογισμό μιας σειράς πράξεων δεν είναι το αναμενόμενο, καθώς αποδόθηκε λάθος η σειρά των πράξεων
	E. έλεγχος της ορθότητας της σύνταξης κάθε εντολής

Μονάδες 10

2.2 Δίνεται ο παρακάτω κώδικας σε Python:

```
for x in range (A, M, B):  
    print x
```

Για καθεμιά από τις παρακάτω περιπτώσεις, να συμπληρώσετε στο γραπτό σας τις τιμές των A, M, B, έτσι ώστε ο αντίστοιχος κώδικας να εμφανίζει στην οθόνη όλους:

1. τους ακεραίους αριθμούς από 1 μέχρι και 80,
2. τους ακεραίους αριθμούς από 50 μέχρι και 20,
3. τους περιττούς ακεραίους αριθμούς από 81 μέχρι και 151,
4. τους ακεραίους αριθμούς από -50 μέχρι και -5 με βήμα 5,
5. τους θετικούς ακεραίους αριθμούς που είναι μικρότεροι του 200 και πολλαπλάσιοι του 7.

Μονάδες 15

ΘΕΜΑ 2

2.1

1. B, E

2. Γ

3. A, Δ

2.2

1. A=1, M=81, B=1

2. A=50, M=19, B=-1

3. A=81, M=152, B=2

4. A=-50, M=0, B=5

5. A=7, M=200, B=7

ΘΕΜΑ 4

Ένας ιδιοκτήτης βιβλιοπωλείου επισκέπτεται το διαδίκτυο προκειμένου να βρει και να αγοράσει σπάνια κόμικς. Προτίθεται να ξοδέψει μέχρι 300 ευρώ. Να αναπτύξετε πρόγραμμα σε Python το οποίο:

4.1 Να διαβάζει για κάθε κόμικ, την τιμή και τη χώρα προέλευσής του και να επιτρέπει την αγορά του, εφόσον η τιμή του δεν υπερβαίνει το διαθέσιμο υπόλοιπο χρημάτων. Διαφορετικά, να τερματίζει εμφανίζοντας το μήνυμα στην οθόνη «Τέλος αγορών». Να θεωρήσετε ότι όλες οι καταχωρήσεις των δεδομένων είναι σωστές.

Μονάδες 10

4.2 Να εμφανίζει στην οθόνη, το πλήθος των ελληνικών κόμικς (χώρα προέλευσης 'ΕΛΛΑΔΑ' ή 'Ελλάδα') και το πλήθος των ξένων κόμικς που αγόρασε ο ιδιοκτήτης βιβλιοπωλείου.

Μονάδες 5

4.3 Να εμφανίζει στην οθόνη το μεγαλύτερο ποσό που ξόδεψε για την αγορά ενός κόμικ (ακριβότερο κόμικ). Να θεωρήσετε ότι μόνο ένα κόμικ έχει τη μεγαλύτερη τιμή.

Μονάδες 5

4.4 Να εμφανίζει στην οθόνη, το ποσό που περίσσεψε, εφόσον αυτό υπάρχει, διαφορετικά το μήνυμα «Εξαντλήθηκε όλο το ποσό». Να θεωρήσετε ότι πραγματοποιήθηκε τουλάχιστον μια αγορά.

Μονάδες 5

Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1  # Αρχικοποίηση μεταβλητών
2  poso=300
3  total=0
4  total_greek=0
5  max=0
6  synolo=0
7  # 4.1
8  agora = int(input("Τιμή αγοράς κόμικ: "))
9  # Έλεγχος αν η τιμή δεν υπερβαίνει το διαθέσιμο υπόλοιπο χρημάτων
10 while poso-agora>=0:
11     # Πλήθος κόμικς
12     total+=1
13     # Συνολικό ποσό
14     synolo=synolo+agora
15     print "Μπορείτε να το αγοράσετε!"
16     poso=poso-agora
17     xora=raw_input("Εισάγετε χώρα προέλευσης: ")
18     # Έλεγχος χώρας ΕΛΛΑΔΑ ή Ελλάδα
19     if xora=='ΕΛΛΑΔΑ' or xora=='Ελλάδα':
20     # 4.2
21         total_greek+=1
22     # 4.3 Εύρεση μεγαλύτερου ποσού
23         if agora>max:
24             max=agora
25     agora = int(raw_input("Τιμή αγοράς νέου κόμικ: "))
26     print "Τέλος αγορών"
27     print "Το πλήθος των ελληνικών κόμικς είναι: ", total_greek
28     print "Το πλήθος των ξένων κόμικς είναι: ", total-total_greek
29     print "Το ακριβότερο κόμικ είχε τιμή: " , max
30     # 4.4
31     if 300-synolo>0:
32         print "Έχει περισσέψει το ποσό: ", 300-synolo
33     else:
34         print "Εξαντλήθηκε όλο το ποσό"
```

ΘΕΜΑ 2

2.1 Να συμπληρώσετε στο γραπτό σας τα αριθμημένα κενά της στήλης **A**, έτσι ώστε κάθε ένα από παρακάτω τμήματα εντολών σε Python να εμφανίζει στην οθόνη το αποτέλεσμα της στήλης **B**

Στήλη A – Τμήμα εντολών σε Python	Στήλη B – Αποτέλεσμα στην οθόνη
<pre>sampleList = [10, 20, 30, 40, 50] sampleList.append(__ (1) __) print sampleList</pre>	[10, 20, 30, 40, 50, 60]
<pre>x = 8 / 4 * (3 + 2) * 4 __ (2) __ 2 print x</pre>	42
<pre>str = "Πύθωνας" print str[__ (3) __]</pre>	ύθ
<pre>salary = 8000 def printsalary(): salary= __ (4) __ print "Αποδοχές: ", salary printsalary () print "Αποδοχές: ", salary</pre>	Αποδοχές: 12000 Αποδοχές: 8000
<pre>items = ["Chair", "Table"] for y in __ (5) __: print y</pre>	Chair Table

Μονάδες 10

2.2 Το παρακάτω πρόγραμμα σε Python έχει γραφεί για να δέχεται από τον χρήστη έναν ακέραιο αριθμό **n**. Στην περίπτωση που ο αριθμός **n** είναι θετικός, να δέχεται από τον χρήστη **n** θετικούς ακέραιους αριθμούς και να εμφανίζει στην οθόνη τον μεγαλύτερο από αυτούς. Οι εντολές του προγράμματος, όμως, δεν δίνονται στη σωστή σειρά. Να γράψετε στο γραπτό σας τον κώδικα με τις εντολές τοποθετημένες στη σωστή σειρά, εισάγοντας τις απαραίτητες εσοχές, ώστε να εκτελείται σωστά.


```
1  for i in range(1,plithos+1):
2  plithos=int(input("Εισάγετε πλήθος: "))
3  max=-1
4  if (plithos>0):
5  if (num>max):
6  max=num
7  num=int(input("Εισάγετε ακέραιο αριθμό: "))
8  print "Ο μεγαλύτερος αριθμός είναι: ", max
```

Μονάδες 15

ΘΕΜΑ 2

2.1

1. – (60)

2. – (+)

3. – (1:3)

4. – (12000)

5. – (items)

2.2

```
2  plithos = int(input("Εισάγετε πλήθος: "))
4  if (plithos>0):
3      max = -1
1      for i in range(1, plithos+1):
7          num = int(input("Εισάγετε ακέραιο αριθμό: "))
5          if (num > max):
6              max = num
8  print "Ο μεγαλύτερος αριθμός είναι: ", max
```

ΘΕΜΑ 4

Για την πρώτη φάση της Ολυμπιάδας Πληροφορικής δήλωσαν συμμετοχή 500 μαθητές. Οι μαθητές διαγωνίζονται σε τρεις γραπτές δοκιμασίες που βαθμολογούνται με ακέραιους βαθμούς στη βαθμολογική κλίμακα από 0 έως και 100. Στην επόμενη φάση της Ολυμπιάδας περνάνε οι μαθητές με μέσο όρο στις τρεις δοκιμασίες μεγαλύτερο του 80. Να γράψετε πρόγραμμα σε Python το οποίο:

4.1 Να διαβάζει για κάθε μαθητή το όνομα και τις βαθμολογίες στις τρεις γραπτές δοκιμασίες. Απαιτείται έλεγχος εγκυρότητας δεδομένων για τις τιμές των βαθμολογιών (κλίμακα από 0 έως 100).

Μονάδες 8

4.2 Να υπολογίζει τον μέσο όρο των τριών βαθμολογιών για κάθε μαθητή και εφόσον είναι μεγαλύτερος του 80 να αποθηκεύει το όνομα του μαθητή σε μια λίστα `names[]` και τον μέσο όρο του σε μια λίστα `medium[]`. Διαφορετικά, να εμφανίζει στην οθόνη μήνυμα «Ο μαθητής NAME δεν πέρασε στην επόμενη φάση», όπου NAME το όνομα του συγκεκριμένου μαθητή.

Μονάδες 9

4.3 Να εμφανίζει στην οθόνη τον αριθμό των μαθητών που πέρασαν στην επόμενη φάση. Να θεωρήσετε ότι ένας τουλάχιστον μαθητής πέρασε στην επόμενη φάση.

Μονάδες 3

4.4 Να εμφανίζει στην οθόνη για κάθε μαθητή που πέρασε στην επόμενη φάση, το όνομά του, ακολουθούμενο από τον μέσο όρο του.

Μονάδες 5

Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1  # Δημιουργία λιστών
2  names=[]
3  medium=[]
4  # 4.1
5  for i in range(500):
6      print "\n"
7      name=raw_input("Εισάγετε όνομα μαθητή: ")
8      sum=0
9      # Έλεγχος εγκυρότητας δεδομένων
10     for k in range(1, 4):
11         b=int(input("Εισάγετε βαθμολογία (0-100): "))
12         while not(b>=0 and b<=100):
13             b=int(input("Προσοχή στην κλίμακα! Εισάγετε βαθμολογία (0-100): "))
14     # 4.2 Υπολογισμός μέσου όρου
15     sum=sum+b
16     m_o=sum/3
17     print "Ο μέσος όρος της βαθμολογίας είναι: ", m_o
18     # Έλεγχος αν η μέσος όρος >80
19     if m_o<=80:
20         print "Ο μαθητής ", name, " δεν πέρασε στην επόμενη φάση"
21     else:
22         # Προσθήκη στοιχείων στις λίστες
23         names.append(name)
24         medium.append(m_o)
25     lnames=len(names)
26     # 4.3
27     print "\n"
28     print "Ο αριθμός των μαθητών που πέρασαν στην επόμενη φάση είναι: ", lnames
29     # 4.4
30     print "\n"
31     print "Τα στοιχεία των μαθητών που πέρασαν στην επόμενη φάση είναι: "
32     for i in range(lnames):
33         print names[i], medium[i]
```

ΘΕΜΑ 2

2.1 Να αντιστοιχήσετε κάθε ένα στοιχείο της στήλης **A** με ένα στοιχείο της στήλης **B**, λαμβάνοντας υπόψη ότι δύο (2) στοιχεία της στήλης **B** περισσεύουν

Στήλη A – Δομές δεδομένων στην Python	Στήλη B - Ορισμός
1. Λίστες 2. Λεξικά 3. Σύνολα 4. Συμβολοσειρές 5. Πλειάδες	A. ακολουθία από χαρακτήρες που μπορεί να αποτελείται από περισσότερες από μία λέξεις, με τις λέξεις να μπορούν να είναι στην Ελληνική Γλώσσα, στην Αγγλική ή σε κάθε γλώσσα που υποστηρίζεται από το πρότυπο Unicode
	B. ομάδα από μη διατεταγμένα αντικείμενα, με κάθε αντικείμενο να εμφανίζεται μία φορά
	Γ. σύνθετος τύπος του οποίου τα στοιχεία είναι αμετάβλητα
	Δ. η θέση κάθε στοιχείου δεν είναι μοναδική και μπορεί να περιέχει ταυτόχρονα δυο τιμές
	Ε. διατεταγμένη ακολουθία από αντικείμενα τα οποία συνήθως είναι του ίδιου τύπου
	ΣΤ. σύνολο ζευγών κλειδιών-τιμών, όπου κάθε κλειδί δεν εμφανίζεται δεύτερη φορά
	Ζ. δέχονται δεδομένα μέσω των παραμέτρων και επιστρέφουν τα αποτελέσματα μέσω άλλων ή και των ίδιων παραμέτρων στο πρόγραμμα

Μονάδες 10

2.2 Δίνεται ο παρακάτω κώδικας σε Python:

```

1 def merge_list(list1, list2):
2     result_list = []
3     for num in list1:
4         if num % 2 != 0:
5             result_list.append(num)
6     for num in list2:
7         if num % 2 == 0:
8             result_list.append(num)
9     return result_list
10 list1 = [10, 20, 25, 30, 35]
11 list2 = [40, 45, 60, 75, 90]
12 print "result list:", merge_list(list1, list2)

```

Να γράψετε στο γραπτό σας:

1. τι θα εμφανιστεί στην οθόνη κατά την εκτέλεση του κώδικα, και
2. να συμπληρώσετε τον παρακάτω πίνακα με τα στοιχεία του κώδικα που ζητάει:

Μεταβλητές	
Συναρτήσεις	
Δομές δεδομένων	
Υπαρξιακοί τελεστές	
Λογικές εκφράσεις	
Αριθμητικοί τελεστές	
Μέθοδοι	
Λογικοί τελεστές σύγκρισης	
Δομές επανάληψης	
Δομές επιλογής	

Μονάδες 15

ΘΕΜΑ 2

2.1

1. Ε

2. ΣΤ

3. Β

4. Α

5. Γ

2.2

1. result list: [25, 35, 40, 60, 90]

2.

Μεταβλητές	num
Συναρτήσεις	merge_list(list1, list2)
Δομές δεδομένων	result_list
Υπαρξιακοί τελεστές	in
Λογικές εκφράσεις	num % 2 != 0 num % 2 == 0
Αριθμητικοί τελεστές	%
Μέθοδοι	append()
Λογικοί τελεστές σύγκρισης	!= ==
Δομές επανάληψης	for num in list1 for num in list2
Δομές επιλογής	if num % 2 != 0 if num % 2 == 0

ΘΕΜΑ 4

Θα υλοποιήσουμε το παιχνίδι Κορώνα – Γράμματα σε προγραμματιστικό περιβάλλον, όπου ο παίκτης που "ρίχνει" το νόμισμα είναι πάντα ο υπολογιστής. Αντιστοιχίζουμε τον αριθμό 0 στην επιλογή ΚΟΡΩΝΑ και τον αριθμό 1 στην επιλογή ΓΡΑΜΜΑΤΑ. Το ρίξιμο του νομίσματος ηλεκτρονικά γίνεται με τη χρήση συνάρτησης παραγωγής τυχαίων αριθμών `random()`. Να αναπτύξετε πρόγραμμα σε Python το οποίο:

4.1 Να εμφανίζει στην οθόνη το μήνυμα «Θα παίξουμε το παιχνίδι ΚΟΡΩΝΑ - ΓΡΑΜΜΑΤΑ», ζητάει το όνομα του παίκτη και επαναληπτικά την επιλογή του: Κ για ΚΟΡΩΝΑ ή Γ για ΓΡΑΜΜΑΤΑ. Το παιχνίδι τερματίζεται όταν ο παίκτης πληκτρολογήσει οτιδήποτε άλλο εκτός από Κ ή Γ.

Μονάδες 5

4.2 Να εμφανίζει στην οθόνη την επιλογή του παίκτη «Επέλεξε ΚΟΡΩΝΑ» ή «Επέλεξε ΓΡΑΜΜΑΤΑ» και το αποτέλεσμα της ηλεκτρονικής ρίψης του νομίσματος «Το νόμισμα έπεσε ΚΟΡΩΝΑ» ή «Το νόμισμα έπεσε ΓΡΑΜΜΑΤΑ». Στη συνέχεια, εμφανίζει το μήνυμα «Μπράβο! Κέρδισες» ή «Λυπάμαι... Έχασες» ανάλογα με την επιλογή του παίκτη και το αποτέλεσμα της ηλεκτρονικής ρίψης.

Μονάδες 5

4.3 Να εμφανίζει στην οθόνη, τον συνολικό αριθμό των ρίψεων, τον αριθμό των ρίψεων που κέρδισε ο παίκτης και τον αριθμό των ρίψεων που έχασε ο παίκτης.

Μονάδες 10

4.4 Να εμφανίζει στην οθόνη, το μήνυμα «Μπράβο NAME τις περισσότερες φορές κέρδισες!» όπου NAME το όνομα του παίκτη, αν ο παίκτης έχει μεγαλύτερο αριθμό ρίψεων που κέρδισε σε σχέση με αυτές που έχασε.

Μονάδες 5

Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1 import random
2 # 4.1
3 print "Θα παίξουμε το παιχνίδι ΚΟΡΩΝΑ - ΓΡΑΜΜΑΤΑ"
4 name=raw_input("Πως σε λένε; ")
5 # Αρχικοποίηση μεταβλητών
6 met=nikes=0
7 choice=raw_input("Τι επιλέγεις; ΚΟΡΩΝΑ (Κ) ή ΓΡΑΜΜΑΤΑ (Γ); ")
8 while choice=="Κ" or choice=="Γ":
9 # Αριθμός ρίψεων
10     met+=1
11 # 4.2
12     if choice=="Κ":
13         print "Επέλεξες ΚΟΡΩΝΑ"
14     else:
15         print "Επέλεξες ΓΡΑΜΜΑΤΑ"
16 # Ηλεκτρονική ρίψη νομίσματος
17     number = random.randint(0,1)
18     if number==0:
19         print "Το νόμισμα έπεσε ΚΟΡΩΝΑ"
20     else:
21         print "Το νόμισμα έπεσε ΓΡΑΜΜΑΤΑ"
22     if (choice=="Κ" and number==0) or (choice=="Γ" and number==1):
23         print "Μπράβο! Κέρδισες"
24         nikes+=1
25     else:
26         print "Λυπάμαι... Έχασες"
27     print "\n"
28     choice=raw_input("Τι επιλέγεις; ΚΟΡΩΝΑ (Κ) ή ΓΡΑΜΜΑΤΑ (Γ); ")
29 print "\n"
30 print "Το παιχνίδι τελείωσε!"
31 # 4.3
32 print name, "έπαιξες", met, "παιχνίδια"
33 print "Κέρδισες", nikes, "φορές"
34 print "Έχασες", met-nikes, "φορές"
35 # 4.4
36 if nikes>met-nikes:
37     print "Μπράβο",name,"τις περισσότερες φορές κέρδισες"
```

ΘΕΜΑ 2

2.1. Δίνεται το παρακάτω πρόγραμμα σε Python:

```
1      X= 0
2      A=1
3      N=0
4      M=2
5      while A < 5:
6          X = X + A
7          if X % 2 == 0:
8              N = N + 1
9          else:
10             M = M + 1
11         A= A + 2
12     print "X= ", X
```

Επίσης, δίνεται το ακόλουθο υπόδειγμα πίνακα τιμών:

Αριθμός εντολής	X	A	N	M
1	0			
...	...	...	...	...

Να μεταφέρετε στο γραπτό σας τον πίνακα τιμών και να τον συμπληρώσετε, δημιουργώντας μια νέα γραμμή στον πίνακα για κάθε εντολή εκχώρησης που εκτελείται. Σε κάθε γραμμή του πίνακα θα γράφετε στην πρώτη στήλη τον αριθμό της εντολής εκχώρησης που εκτελείται και τη νέα τιμή της μεταβλητής που τροποποιείται στην αντίστοιχη στήλη.

Μονάδες 15

2.2 Να συμπληρώσετε τα αριθμημένα κενά στο τμήμα εντολών σε Python στη στήλη Β, χωρίς να χρησιμοποιήσετε λογικούς τελεστές, ώστε να είναι ισοδύναμο με το τμήμα εντολών της στήλης Α.

Στήλη Α	Στήλη Β
A=int(input("Type an integer: ")) B=int(input("Type an integer: ")) if not(A==B): C = A * B elif A>=5 and A != 5: C = A + B print C	A=int(input("Type an integer: ")) B=int(input("Type an integer: ")) if <u> (1) </u> : C = A * B elif <u> (2) </u> : C = A + B print C

Μονάδες 10

ΘΕΜΑ 2

2.1

Αριθμός εντολής	X	A	N	M
1	0			
2		1		
3			0	
4				2
6	1			
10				3
11		3		
6	4			
8			1	
11		5		

2.2

(1): $A \neq B$

(2): $A > 5$

ΘΕΜΑ 4

Μια αλυσίδα ξενοδοχείων διαθέτει 15 ξενοδοχεία, άλλα σε μεγαλουπόλεις κι άλλα σε νησιά. Λόγω της πανδημίας του COVID-19, τα έσοδα της αλυσίδας μειώθηκαν σημαντικά και πρέπει να κλείσει ένα ξενοδοχείο, αυτό με το χαμηλότερο ετήσιο κέρδος, ώστε να περιορίσει την οικονομική ζημιά. Για το σκοπό αυτό συγκέντρωσε τα οικονομικά στοιχεία του περασμένου έτους και για τα 15 ξενοδοχεία, ώστε να τα επεξεργαστεί στατιστικά.

Να αναπτυχθεί πρόγραμμα σε Python το οποίο:

4.1 Να διαβάζει το όνομα κάθε ξενοδοχείου, τις ετήσιες εισπράξεις του, τα ετήσια έξοδά του και την τοποθεσία του. Οι εισπράξεις και τα έξοδα δίνονται σε ευρώ. Η τοποθεσία είναι 'ΠΟΛΗ' για τα ξενοδοχεία που βρίσκονται σε πόλη και 'ΝΗΣΙ' για τα ξενοδοχεία που βρίσκονται σε νησί. Να γίνεται έλεγχος της τιμής που εισάγεται για την τοποθεσία ως προς την εγκυρότητά της. Στην περίπτωση που πληκτρολογείται μη επιτρεπτή τιμή, το πρόγραμμα να ζητάει νέα τιμή μέχρι να δοθεί έγκυρη.

Μονάδες 7

4.2 Να υπολογίζει το ετήσιο κέρδος ή ζημία του κάθε ξενοδοχείου (ζημία είναι το αρνητικό κέρδος), αφαιρώντας από τα ετήσια έσοδα τα ετήσια έξοδα. Αν το ξενοδοχείο είχε κέρδος θα το εμφανίζει μετά από το μήνυμα "ΚΕΡΔΟΣ :", διαφορετικά θα εμφανίζει το μήνυμα "ΖΗΜΙΑ:" ακολουθούμενο από την απόλυτη τιμή της ζημίας. Θεωρήστε ότι τα ετήσια έσοδα δε μπορεί να είναι ίσα με τα ετήσια έξοδα.

Μονάδες 5

4.3 Να υπολογίζει και να εμφανίζει στην οθόνη, με κατάλληλο μήνυμα, το πλήθος των ξενοδοχείων της εταιρίας που βρίσκονται σε νησί και κατά το περασμένο έτος εμφάνισαν ζημία.

Μονάδες 6

4.4 Να εντοπίζει το ξενοδοχείο με το μικρότερο ετήσιο κέρδος. Να εμφανίζει στην οθόνη το μήνυμα "ΘΑ ΚΛΕΙΣΕΙ ΤΟ ΞΕΝΟΔΟΧΕΙΟ ", ακολουθούμενο από το όνομα του ξενοδοχείου.

Μονάδες 7

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1  MIN_PROFIT = 10000000. # 4.4 Αρχικοποίηση του ελάχιστου κέρδους
2  PL= 0 # 4.3 Αρχικοποίηση του πλήθους των ξενοδοχείων σε νησί με ζημία
3
4  # 4.1: Εισαγωγή των δεδομένων για τα 15 ξενοδοχεία
5  for i in range(15):
6      NAME= raw_input('Δώσε το όνομα του ξενοδοχείου:')
7      ESODA = float(input('Δώσε τα ετήσια έσοδά του σε ευρώ:'))
8      EXPENSES = float(input('Δώσε τα ετήσια έξοδά του σε ευρώ:'))
9      TOPOS= raw_input('Δώσε τοποθεσία (ΠΟΛΗ για πόλη ή ΝΗΣΙ για νησί:'))
10 # 4.1 έλεγχος εγκυρότητας των δεδομένων που εισάγονται στην τοποθεσία
11     while TOPOS != 'ΠΟΛΗ' and TOPOS != 'ΝΗΣΙ':
12         print 'Άκυρη τοποθεσία!'
13         TOPOS= raw_input('Δώσε ΠΟΛΗ για πόλη ή ΝΗΣΙ για νησί:')
14 # 4.2
15     PROFIT= ESODA - EXPENSES
16     if PROFIT >= 0:
17         print 'ΚΕΡΔΟΣ: ', PROFIT
18     else:
19         print 'ΖΗΜΙΑ: ', abs(PROFIT)
20
21 # 4.3
22     if TOPOS == 'ΝΗΣΙ' and PROFIT < 0:
23         PL = PL + 1
24
25 # 4.4: αν το κέρδος του είναι μικρότερο από το τρέχον ελάχιστο,
26 # ενημερώνεται και η ελάχιστη τιμή, αλλά και το όνομα
27 # του ξενοδοχείου που θα κλείσει
28     if PROFIT < MIN_PROFIT:
29         MIN_PROFIT = PROFIT
30         MIN_NAME = NAME
31
32 # 4.3
33     print 'Πλήθος των ξενοδοχείων σε νησί που είχαν ζημία: ', PL
34 # 4.4
35     print 'ΘΑ ΚΛΕΙΣΕΙ ΤΟ ΞΕΝΟΔΟΧΕΙΟ: ', MIN_NAME
```


ΘΕΜΑ 2

2.1.

A. Να γράψετε ισοδύναμο πρόγραμμα με το παρακάτω πρόγραμμα σε Python, χρησιμοποιώντας την εντολή επανάληψης while αντί της εντολής επανάληψης for.

```
1 sum=0
2 for x in range(10):
3     if x % 2 == 1 :
4         sum = sum + x
5     print "Το άθροισμα είναι:", sum
```

B. Πόσες φορές θα εκτελεστεί η εντολή 4 ($sum = sum + x$) του αρχικού προγράμματος και τι θα εμφανιστεί στην οθόνη κατά τη διάρκεια εκτέλεσής του;

Μονάδες 10

2.2.

Στο παρακάτω τμήμα προγράμματος σε Python υπάρχουν πέντε (5) αριθμημένα κενά. Να γράψετε στο γραπτό σας τον αριθμό του κάθε κενού και δίπλα τι πρέπει να συμπληρωθεί, ώστε το τμήμα προγράμματος να υπολογίζει και να εμφανίζει το γινόμενο των άρτιων ακεραίων από το 2 έως και το 50.

```
1 ginomeno= ____(1)
2 for x in range( ____(2), ____(3), ____(4)) :
3     ginomeno= ____(5)
4     print "Το γινόμενο είναι:", ginomeno
```

Μονάδες 15

ΘΕΜΑ 2

2.1

A. Το ισοδύναμο πρόγραμμα με χρήση της εντολής επανάληψης while είναι το εξής:

```
1  sum=0
2  x=0
3  while x < 10:
4      if x % 2 == 1 :
5          sum= sum + x
6      x = x+1
7  print "Το άθροισμα είναι:", sum
```

B. Η εντολή 4 (sum= sum + x) του αρχικού προγράμματος θα εκτελεστεί **5 φορές**.

Κατά την διάρκεια της εκτέλεσης του προγράμματος θα εμφανιστεί στην οθόνη ο αριθμός **25**.

2.2

(1): 1
(2): 2
(3): 52
(4): 2
(5): ginomeno * x (ή x * ginomeno)

ΘΕΜΑ 4

Κατά την είσοδο των οχημάτων στον αυτοκινητόδρομο της Αττικής οδού οι οδηγοί πληρώνουν διόδια στον πλησιέστερο σταθμό διοδίων. Η τιμή χρέωσης για κάθε όχημα εξαρτάται από την κατηγορία στην οποία ανήκει: Οι οδηγοί δικύκλων (κατηγορία 1) πληρώνουν 1.40 ευρώ, οι οδηγοί επιβατηγών και μικρών εμπορικών οχημάτων (κατηγορία 2) πληρώνουν 2.80 ευρώ, οι οδηγοί μεσαίων φορτηγών και μεγάλων λεωφορείων (κατηγορία 3) πληρώνουν 7.10 ευρώ και οι οδηγοί μεγάλων φορτηγών (κατηγορία 4) πληρώνουν 11.30 ευρώ. Ένα κλιμάκιο της εταιρίας διαχείρισης του αυτοκινητοδρόμου της Αττικής οδού βρίσκεται σε έναν σταθμό διοδίων και καταγράφει για στατιστικούς λόγους στοιχεία για τις διελεύσεις μίας ημέρας. Να γραφεί πρόγραμμα σε γλώσσα προγραμματισμού Python που:

4.1 Για κάθε όχημα που διέρχεται από τον σταθμό των διοδίων τη συγκεκριμένη ημέρα διαβάζει τον αριθμό της πινακίδας του και την κατηγορία του (1-4). Να ελέγχεται η τιμή που εισάγεται για την κατηγορία ως προς την ορθότητά της. Στην περίπτωση λάθους να επανεισάγεται μέχρι να δοθεί ορθή τιμή. Η εισαγωγή των δεδομένων ολοκληρώνεται όταν δίνεται ως αριθμός πινακίδας η λέξη “ΤΕΛΟΣ”.

Μονάδες 9

4.2 Για κάθε όχημα που καταχωρείται εμφανίζει τον αριθμό πινακίδας του οχήματος, ακολουθούμενο από το ποσό που πρέπει να πληρώσει ο οδηγός του.

Μονάδες 8

4.3 Υπολογίζει κι εμφανίζει το συνολικό ποσό είσπραξης από όλες τις διελεύσεις της ημέρας, καθώς και το πλήθος των φορτηγών και λεωφορείων (οχημάτων που ανήκουν στις κατηγορίες 3 και 4) που διήλθαν από τον σταθμό διοδίων την ημέρα αυτή.

Μονάδες 8

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1  sum = 0 # 4.3 Αρχικοποίηση συνολικής είσπραξης ημέρας
2  pl = 0 # 4.3 Αρχικοποίηση του πλήθους των οχημάτων κατηγορίας 3 ή 4
3
4  # Εισαγωγή πινακίδας 1ου οχήματος
5  ar_kykl = raw_input(u'Δώσε τον αριθμό κυκλοφορίας του 1ου οχήματος:')
6
7  # 4.1: η εισαγωγή των δεδομένων συνεχίζεται όσο
8  # δεν έχει δοθεί ως πινακίδα η λέξη "ΤΕΛΟΣ"
9  while ar_kykl != u'ΤΕΛΟΣ':
10     cat = int(input(u'Δώσε την κατηγορία του οχήματος (1-4):'))
11     # 4.1 έλεγχος εγκυρότητας των δεδομένων που εισάγονται στην κατηγορία
12     while cat < 1 or cat > 4:
13         print u'Άκυρη τιμή !'
14         cat = int(input(u'Δώσε ξανά την κατηγορία του οχήματος (1-4):'))
15
16     # 4.2 Υπολογισμός κι εμφάνιση ποσού χρέωσης οχήματος
17     if cat == 1:
18         xreosi = 1.4
19     elif cat == 2:
20         xreosi = 2.8
21     elif cat == 3:
22         xreosi = 7.10
23     else:
24         xreosi = 11.30
25     print u'Το όχημα με πινακίδα ', ar_kykl, u' οφείλει', xreosi, u' ευρώ.'
26
27     # 4.3: ενημέρωση συνολικής είσπραξης ημέρας
28     sum = sum + xreosi
29     # 4.3: αν πρόκειται για όχημα κατηγορίας 3 ή 4
30     if cat == 3 or cat == 4:
31         pl = pl + 1
32
33     # Εισαγωγή πινακίδας επόμενου οχήματος
34     ar_kykl = raw_input(u'Δώσε την πινακίδα του επόμενου οχήματος:')
35
36     # 4.3 Αποτελέσματα
37     print u'Συνολικό ποσό είσπραξης της ημέρας (σε ευρώ):', sum
38     print u'Πλήθος οχημάτων κατ. 3 ή 4: ', pl
```

ΘΕΜΑ 2

2.1 Να αντιστοιχήσετε κάθε ένα από τα στοιχεία της στήλης **A** με ένα στοιχείο της στήλης **B**, λαμβάνοντας υπόψη ότι ένα (1) στοιχείο της στήλης **B** περισσεύει

Στήλη A – Προγραμματιστικά Υποδείγματα	Στήλη B - Περιγραφή
1. Παράλληλος προγραμματισμός	A. παρέχει τη δυνατότητα δημιουργίας του προγράμματος μέσω γραφικών αντικειμένων, αντί της πληκτρολόγησης του κειμένου που αντιστοιχεί σε εντολές
2. Οπτικός προγραμματισμός	B. το πρόγραμμα είναι ένα σύνολο από αξιώματα ή κανόνες οι οποίοι καθορίζουν σχέσεις ανάμεσα σε αντικείμενα
3. Προγραμματισμός οδηγούμενος από γεγονότα	Γ. το πρόβλημα διαιρείται σε απλούστερα τμήματα, αυτά με τη σειρά τους σε επί μέρους μικρότερα κ.ο.κ.
4. Λογικός προγραμματισμός	Δ. επιτρέπει ταυτόχρονη εκτέλεση διαδικασιών από διαφορετικούς επεξεργαστές
5. Αρθρωτός ή Τμηματικός Προγραμματισμός	E. η ροή του προγράμματος εξαρτάται από την ύπαρξη γεγονότων (events)
	ΣΤ. δημιουργία μικρών τμημάτων κώδικα και όχι ολοκληρωμένων προγραμμάτων

Μονάδες 10

2.2 Δίνονται τα παρακάτω δύο τμήματα κώδικα σε Python. Να ξαναγράψετε στο γραπτό σας τα παρακάτω τμήματα μετατρέποντας:

1. τη δομή επανάληψης for στη δομή επανάληψης while

```
1 a=0
2 for k in range(10, 0, -2):
3     a=a+k**2
4     print a
```

2. τη δομή επανάληψης while στη δομή επανάληψης for

```
1 s=0
2 k=5
3 while k>1:
4     s+=k
5     print k
6     k-=1
7 print s
```

Μονάδες 15

ΘΕΜΑ 2

2.1

1 – Δ

2 – Α

3 – Ε

4 – Β

5 – Γ

2.2

1.

```
1 a=0
2 k=10
3 while k>0:
4     a=a+k**2
5     k-=2
6     print a
```

2.

```
1 s=0
2 for k in range(5, 1, -1):
3     s+=k
4     print k
5 print s
```


ΘΕΜΑ 4

Το Υπουργείο Εσωτερικών για τις ανάγκες των εξετάσεων για την απόκτηση ιθαγένειας έχει δεσμεύσει 150 αίθουσες διαφορετικής χωρητικότητας σε σχολικά συγκροτήματα σε ολόκληρη τη χώρα. Ο αριθμός των επιτηρητών που απαιτούνται ανά αίθουσα, καθορίζεται από το πλήθος των εξεταζόμενων που βρίσκονται σε αυτή, σύμφωνα με τον παρακάτω πίνακα:

Άτομα που εξετάζονται	Αριθμός επιτηρητών ανά αίθουσα
μέχρι και 10	2
από 11 μέχρι και 21	3
περισσότερα από 21	4

Να αναπτύξετε πρόγραμμα σε Python το οποίο:

4.1 Για κάθε αίθουσα να διαβάζει τα ονόματα των εξεταζόμενων που θα βρίσκονται σε αυτήν και να υπολογίζει το πλήθος των επιτηρητών που απαιτούνται για αυτήν. Η διαδικασία της εισαγωγής των ονομάτων για κάθε αίθουσα τερματίζεται μόλις εισαχθεί ως όνομα εξεταζόμενου το "0000". Να θεωρήσετε ότι η καταχώρηση των ατόμων για κάθε αίθουσα είναι σωστή και δεν υπάρχει περίπτωση να παραβιάζεται η χωρητικότητα κάποιας αίθουσας.

Μονάδες 8

4.2 Να εμφανίζει στην οθόνη το πλήθος των επιτηρητών που απαιτούνται για κάθε αίθουσα για τη διενέργεια των εξετάσεων.

Μονάδες 2

4.3 Να εμφανίζει στην οθόνη το πλήθος των εξεταζόμενων και το πλήθος των επιτηρητών που απαιτούνται συνολικά για τη διενέργεια των εξετάσεων.

Μονάδες 8

4.4 Να εμφανίζει στην οθόνη το μεγαλύτερο πλήθος εξεταζόμενων που θα εξεταστούν σε μια από αυτές τις αίθουσες. Να θεωρήσετε ότι το μεγαλύτερο πλήθος είναι μοναδικό.

Μονάδες 7

Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1  # Αρχικοποίηση μεταβλητών
2  syn_epit=0
3  syn_atom=0
4  max=-1
5  # 4.1
6  for k in range (1,151):
7      print "\n"
8      print "Αίθουσα: ", k
9      plithos=0
10     name=raw_input("Εισάγετε όνομα εξεταζόμενου για αυτή την αίθουσα: ")
11     while (name!="0000"):
12         # Πλήθος εξεταζόμενων ανά αίθουσα
13         plithos+=1
14         name=raw_input("Εισάγετε όνομα εξεταζόμενου για αυτή την αίθουσα: ")
15         # Υπολογισμός επιτηρητών για την αίθουσα
16         if plithos <= 10:
17             pl_epit=2
18         elif plithos<=21:
19             pl_epit=3
20         else:
21             pl_epit=4
22         # 4.4 Υπολογισμός μεγαλύτερου πλήθους εξεταζόμενων
23         if plithos > max:
24             max=plithos
25         # 4.2
26         print "Πλήθος απαιτούμενων επιτηρητών για την αίθουσα ", k, " : ", pl_epit
27         # 4.3 Υπολογισμός πλήθους εξεταζόμενων και επιτηρητών συνολικά
28         syn_atom+=plithos
29         syn_epit+=pl_epit
30     print "\n"
31     print "Το πλήθος των εξεταζόμενων συνολικά είναι: ", syn_atom
32     print "\n"
33     print "Το πλήθος των επιτηρητών που απαιτούνται συνολικά είναι: ", syn_epit
34     # 4.4
35     print "\n"
36     print "Το μεγαλύτερο πλήθος εξεταζόμενων σε μια αίθουσα είναι: ", max
```

ΘΕΜΑ 2

2.1 Δίνεται το παρακάτω τμήμα κώδικα σε Python. Να ξαναγράψετε στο γραπτό σας το παρακάτω τμήματα μετατρέποντας τη δομή επανάληψης for σε δομή επανάληψης while:

```
1 for i in range(1,5,2):  
2     print i, i**2
```

Μονάδες 5

2.2 Να γράψετε στο γραπτό σας το αποτέλεσμα το οποίο θα εμφανιστεί στην οθόνη μετά την εκτέλεση του παρακάτω τμήματος προγράμματος σε Python:

```
1 old=[ ]  
2 for x in range(1,6):  
3     old.append(x*(x+1))  
4 print old
```

Μονάδες 5

2.3 Δίνεται το παρακάτω τμήμα κώδικα σε Python:

```
1 p= __(1)____  
2 for k in range( __(2)____ ):  
3     num=input("Εισάγετε αριθμό: ")  
4     p= __(3)____ * __(4)____  
5 print "Γινόμενο: ", __(5)____
```

Να συμπληρώσετε τα αριθμημένα κενά από το (1) μέχρι και το (5), έτσι ώστε το παραπάνω τμήμα κώδικα να διαβάζει 10 θετικούς ακέραιους αριθμούς και να υπολογίζει και να εμφανίζει στην οθόνη το γινόμενό τους.

Μονάδες 15

ΘEMA 2

2.1

```
1 i=1
2 while i<5:
3     print i, i**2
4     i+=2
```

2.2

[2, 6, 12, 20, 30]

2.3

1 – 1

2 – 10

3 - p

4 – num

5 - p

ΘΕΜΑ 4

Κατά τη διάρκεια μιας ανακαίνισης σχολείου πρόκειται να χρησιμοποιηθούν 15 κάδοι με κωδικούς από 1 μέχρι 15 για τη μεταφορά των οικοδομικών υλικών, με κάθε κάδο να μπορεί να μεταφέρει μέχρι 10 τόνους. Να αναπτύξετε πρόγραμμα σε Python το οποίο:

4.1 Να διαβάζει επαναληπτικά το βάρος των οικοδομικών υλικών σε κιλά που δέχεται ο κάθε κάδος μέχρι να εισαχθεί αρνητικός αριθμός ή το μηδέν. Αν ο κάδος περιέχει φορτίο ακριβώς όσο μπορεί να μεταφέρει, να εμφανίζεται το μήνυμα στην οθόνη «Κάδος πλήρης», ενώ αν μπορεί να φορτωθεί επιπλέον βάρος, να εμφανίζεται το μήνυμα στην οθόνη «Επιπλέον βάρος» και το βάρος που μπορεί να φορτωθεί. Να θεωρήσετε ότι κανένας κάδος δεν θα φορτωθεί με βάρος μεγαλύτερο του επιτρεπτού.

Μονάδες 10

4.2. Για κάθε κάδο να εμφανίζει στην οθόνη, το πλήθος των ρίψεων των οικοδομικών υλικών σε αυτόν και το τελικό βάρος που θα μεταφέρει.

Μονάδες 5

4.3 Να εμφανίζει στην οθόνη τους κωδικούς των κάδων που μεταφέρουν το μικρότερο και το μεγαλύτερο βάρος των οικοδομικών υλικών αντίστοιχα, καθώς και το βάρος που μεταφέρει ο καθένας τους. Να θεωρήσετε ότι όλα τα τελικά βάρη των κάδων είναι διαφορετικά.

Μονάδες 10

Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
    print b
```

ΘΕΜΑ 4

```
1  # Αρχικοποίηση μεταβλητών
2  min=11000
3  max=-1
4  min_kodikos=max_kodikos=0
5  # 4.1
6  for k in range (1, 16):
7      ripseis=s_varos=0
8      varos=int(input("Εισάγετε βάρος υλικών σε κιλά: "))
9      while varos>0:
10     # Υπολογισμός βάρους και ρίψεων
11         s_varos+=varos
12         ripseis+=1
13         varos=int(input("Εισάγετε βάρος υλικών: "))
14     if s_varos==10000:
15         print "Κάδος πλήρης"
16     elif s_varos<10000:
17         print "Επιπλέον βάρος: ", 10000-s_varos
18     # 4.3 Υπολογισμός μικρότερου και μεγαλύτερου βάρους
19     if s_varos<min:
20         min=s_varos
21         min_kodikos=k
22     if s_varos>max:
23         max=s_varos
24         max_kodikos=k
25     # 4.2
26     print "Πλήθος ρίψεων: ", ripseis
27     print "Τελικό βάρος στον κάδο: ", s_varos
28     # 4.3
29     print "Ο κάδος ", min_kodikos, "μεταφέρει το μικρότερο βάρος υλικών (Kg): ", min
30     print "Ο κάδος ", max_kodikos, "μεταφέρει το μεγαλύτερο βάρος υλικών (kg):", max
```

ΘΕΜΑ 4

Η εταιρεία ΔΡΟΜΟΣ ΑΕ διαχειρίζεται την είσοδο και τη στάθμευση των οχημάτων στην Πανεπιστημιούπολη Α, με το κόστος να φαίνεται στον παρακάτω πίνακα:

Τύπος οχήματος	Ποσό (ευρώ)
Δίκυκλο	1
Επιβατικό	2
Λεωφορείο	3

Για τις ανάγκες καταγραφής των εισερχόμενων οχημάτων και των εισπράξεων από την εταιρεία, να αναπτύξετε πρόγραμμα σε Python το οποίο:

4.1 Για κάθε εισερχόμενο όχημα να διαβάζει τον τύπο του: "Δ" για τα δίκυκλα, "Ε" για τα επιβατικά και "Λ" για τα λεωφορεία. Απαιτείται έλεγχος εγκυρότητας δεδομένων για τον τύπο του διερχόμενου οχήματος. Όταν δοθεί το "00" ως τύπος οχήματος, σταματά η εισαγωγή δεδομένων.

Μονάδες 10

4.2. Για κάθε εισερχόμενο όχημα να εμφανίζει στην οθόνη το κόστος εισόδου και στάθμευσης του οχήματος αυτού.

Μονάδες 5

4.3 Να εμφανίζει στην οθόνη το πλήθος των οχημάτων κάθε κατηγορίας.

Μονάδες 7

4.4 Να εμφανίζει στην οθόνη το συνολικό ποσό εισπραξης της εταιρείας για όλα τα οχήματα όλων των κατηγοριών.

Μονάδες 3

Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:


```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1  # Αρχικοποίηση μεταβλητών
2  syn_kostos=0
3  syn_ox=0
4  s_D=s_E=s_L=0
5  # 4.1
6  typ=raw_input("Εισάγετε τύπο οχήματος (Ε ή Φ ή Λ): ")
7  # 'Έλεγχος εγκυρότητας δεδομένων'
8  while typ not in ["Δ", "Ε", "Λ"]:
9      typ=raw_input("Προσοχή! Εισάγετε σωστά τον τύπο οχήματος (Δ ή Ε ή Λ): ")
10 while typ!="00":
11 # 4.2 Υπολογισμός κόστους
12     if typ=="Δ":
13         kostos=1
14     elif typ=="Ε":
15         kostos=2
16     else:
17         kostos=3
18 # 4.4 Υπολογισμός συνολικού ποσού είσπραξης
19     syn_kostos+=kostos
20     print "Το κόστος του οχήματος είναι: ", kostos
21 # 4.3 Υπολογισμός πλήθους οχημάτων ανά κατηγορία
22     if typ=="Δ":
23         s_D+=1
24     elif typ=="Ε":
25         s_E+=1
26     else:
27         s_L+=1
28     typ=raw_input("Εισάγετε τύπο οχήματος (Δ ή Ε ή Λ): ")
29 # 'Έλεγχος εγκυρότητας δεδομένων'
30     while typ not in ["Δ", "Ε", "Λ", "00"]:
31         typ=raw_input("Προσοχή! Εισάγετε σωστά τον τύπο οχήματος (Δ ή Ε ή Λ): ")
32 # 4.3
33     print "Το πλήθος των δίκυκλων είναι: ", s_D
34     print "Το πλήθος των επιβατικών είναι: ", s_E
35     print "Το πλήθος των λεωφορείων είναι: ", s_L
36 # 4.4
37     print "Το συνολικό ποσό είσπραξης όλων των οχημάτων κάθε κατηγορίας είναι: ", syn_kostos
```

ΘΕΜΑ 2

2.1 Δίνεται το παρακάτω τμήμα προγράμματος σε Python:

```
1    sum =0
2    x=int(input("Type an integer: "))
3    x=abs(x)
4    for i in range(x):
5        if i % 3 == 0 and i % 2 != 0:
6            sum= sum +i
7    print sum
```

Να γράψετε στο γραπτό σας:

α. Τους αριθμητικούς τελεστές

β. Τους συγκριτικούς τελεστές

γ. Τους λογικούς τελεστές

δ. Τις συνθήκες

ε. Τις ενσωματωμένες συναρτήσεις της Python

που υπάρχουν στο παραπάνω τμήμα προγράμματος.

Μονάδες 10

2.2 Το ακόλουθο πρόγραμμα σε Python έχει γραφεί για να διαβάζει 50 ακέραιους αριθμούς στο διάστημα [1,10] και να υπολογίζει το γινόμενο τους. Στο πρόγραμμα, όμως, υπάρχουν πέντε (5) λάθη. Για κάθε ένα από τα λάθη να γράψετε στο γραπτό σας τον αριθμό της γραμμής στην οποία το εντοπίσατε, να περιγράψετε το λάθος και να δηλώσετε το είδος του λάθους:

```
1    p=0
2    for i in range(1,50):
3        x=integer(input("Δώσε έναν ακέραιο:"))
4        while x < 1 and x > 10:
5            p= p*x
6    print "ΓΙΝΟΜΕΝΟ=" p
```

Μονάδες 15

ΘΕΜΑ 2

2.1

- α. %, +
- β. ==, !=
- γ. and
- δ. $i \% 3 == 0$, $i \% 2 != 0$
- ε. int, abs, range

2.2

Αριθμός γραμμής	Περιγραφή λάθους	Είδος λάθους
1	Το γινόμενο πρέπει να αρχικοποιείται στην τιμή 1 κι όχι στην τιμή 0. <u>Σωστό:</u> p=1	Λογικό
2	Διαβάζονται 49 ακέραιοι κι όχι 50 όπως πρέπει. <u>Σωστό:</u> range(1,51) ή range(50)	Λογικό
3	Η συνάρτηση γράφεται int κι όχι integer. <u>Σωστό:</u> x = int (input("Δώσε έναν ακέραιο:"))	Συντακτικό
4	Δεν μπορεί ένας αριθμός να είναι ταυτόχρονα <1 και >10. Εδώ απαιτείται ο λογικός τελεστής or (κι όχι ο τελεστής and).	Λογικό
6	Λείπει το κόμμα (,) μεταξύ του μηνύματος και της μεταβλητής p	Συντακτικό

ΘΕΜΑ 4

Μια εταιρεία παραγωγής ηλεκτρικής ενέργειας κατά την περασμένη χρονιά εγκατέστησε μονάδες φωτοβολταϊκών συστημάτων σε είκοσι (20) περιοχές της Ελλάδας. Η εταιρεία αποφάσισε να ερευνήσει τις ετήσιες επιδόσεις σε παραγωγή ενέργειας των εγκαταστάσεων αυτών.

Να γραφεί πρόγραμμα σε Python που:

4.1 Διαβάζει το όνομα κάθε περιοχής στην οποία έγινε εγκατάσταση φωτοβολταϊκών συστημάτων κατά το περασμένο έτος και στη συνέχεια διαβάζει την ετήσια παραγωγή ηλεκτρικής ενέργειας σε kWh από τα φωτοβολταϊκά συστήματα στην περιοχή αυτή, ελέγχοντας ότι είναι θετικός αριθμός.

Μονάδες 6

4.2 Υπολογίζει κι εμφανίζει τις συνολικές kWh που παρήχθησαν ετησίως και στις 20 περιοχές μαζί.

Μονάδες 4

4.3 Εμφανίζει το όνομα της περιοχής στην οποία σημειώθηκε η μεγαλύτερη ετήσια παραγωγή ηλεκτρικού ρεύματος, καθώς και της περιοχής στην οποία σημειώθηκε η μικρότερη ετήσια παραγωγή ενέργειας, με κατάλληλα μηνύματα. Θεωρείστε ότι οι περιοχές αυτές είναι μοναδικές. Εάν η μικρότερη ετήσια παραγωγή ενέργειας δεν ξεπέρασε τις 20000 kWh να εμφανίζει μήνυμα «ΔΙΑΚΟΠΤΕΤΑΙ Η ΛΕΙΤΟΥΡΓΙΑ ΦΩΤΟΒΟΛΤΑΪΚΩΝ ΣΤΗΝ ΠΕΡΙΟΧΗ»

Μονάδες 15

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1  sum = 0.0  # 4.2 Αρχικοποίηση συνολ. παραγωγής και στις 20 περιοχές μαζί
2
3  # Ερώτημα 4.1: Εισαγωγή δεδομένων για κάθε περιοχή
4  for x in range(20):
5      nomos = raw_input("Δώσε την περιοχή:")
6      et_par = int(input("Ετήσια παραγωγή ρεύματος στην περιοχή (σε KWh):"))
7      while et_par <= 0:
8          print "Η ετήσια παραγωγή είναι θετικός αριθμός!"
9          et_par = int(input("Δώσε ξανά την ετήσια παραγωγή ρεύματος:"))
10
11 # Ερώτημα 4.2: Υπολογισμός της συνολικής ετήσιας παραγωγής
12 sum = sum + et_par
13
14 # Ερώτημα 4.3
15 if x==0: # αν πρόκειται για την 1η περιοχή που καταχωρείται
16     max_par= et_par # αρχικοποίηση της max παραγωγής
17     max_nomos = nomos # και της αντίστοιχης περιοχής
18     min_par= et_par # αρχικοποίηση της min παραγωγής
19     min_nomos = nomos # και της αντίστοιχης περιοχής
20 else:
21     if et_par > max_par:
22         max_par= et_par # ενημέρωση της max παραγωγής
23         max_nomos = nomos # και της αντίστοιχης περιοχής
24     if et_par < min_par:
25         min_par= et_par # ενημέρωση της min παραγωγής
26         min_nomos = nomos # και της αντίστοιχης περιοχής
27
28 # 4.2:
29 print "Η συνολική παραγωγή ρεύματος και στις 20 περιοχές μαζί: ", sum
30 # 4.3:
31 print "Η περιοχή με τη μεγαλύτερη ετήσια παραγωγή ρεύματος: ", max_nomos
32 print "Η περιοχή με τη μικρότερη ετήσια παραγωγή ρεύματος: ", min_nomos
33 if min_par <= 20000:
34     print "ΔΙΑΚΟΠΤΕΤΑΙ Η ΛΕΙΤΟΥΡΓΙΑ ΦΩΤΟΒΟΛΤΑΪΚΩΝ ΣΤΗΝ ΠΕΡΙΟΧΗ"
```

ΘΕΜΑ 2

2.1 Για κάθε ένα από τα παρακάτω τμήματα εντολών σε Python της στήλης **A** να συμπληρώσετε το αποτέλεσμα που θα εμφανιστεί στην οθόνη στη στήλη **B**

Στήλη A – Τμήμα εντολών σε Python	Στήλη B – Αποτελέσματα στην οθόνη
1 <code>x, y, z = "Ιούνιος", "Ιούλιος", 2 print x, y*z</code>	
2 <code>a=13 a=b="Μαρία" print a, b</code>	
3 <code>x, y=20, 30 x, y=y, x print x, y</code>	
4 <code>K=[1,4,5,6] for a in K: print a*2</code>	
5 <code>sxoleio=[] for a in range(1,5): sxoleio.append(a*(a+1)) print sxoleio</code>	

Μονάδες 10

2.2 Δίνεται το παρακάτω πρόγραμμα σε Python, το οποίο διαβάζει ηλικίες και υπολογίζει και εμφανίζει στην οθόνη τον μέσο όρο τους. Το πρόγραμμα τερματίζεται, όταν πληκτρολογηθεί αριθμός μικρότερος ή ίσος του μηδενός. Το πρόγραμμα έχει σφάλματα.

```
1  s = i = 1.0
2  age = int(input ("Εισάγετε ηλικία: "))
3  while age >= 0:
4      s = s + i
5      i +=1
6      age =int( input ("Εισάγετε ηλικία: "))
7  if i != 0
8      mo = age / i
9      print "Μέσος όρος ηλικίας: ", mo
10 else:
11     print "Δεν δόθηκαν ηλικίες"
```

Να εντοπίσετε τα σφάλματα αυτά και να συμπληρώσετε για κάθε σφάλμα μια ξεχωριστή γραμμή στον πίνακα που ακολουθεί:

Αριθμός γραμμής	Σφάλμα	Διόρθωση σφάλματος	Κατηγορία σφάλματος
--------------------	--------	--------------------	---------------------

Μονάδες 15

ΘΕΜΑ 2

Στήλη Α – Τμήμα εντολών σε Python	Στήλη Β – Αποτελέσματα στην οθόνη
1 <code>x, y, z = "Ιούνιος", "Ιούλιος", 2</code> <code>print x, y*z</code>	Ιούνιος ΙούλιοςΙούλιος
2 <code>a=13</code> <code>a=b="Μαρία"</code> <code>print a, b</code>	Μαρία Μαρία
3 <code>x, y=20, 30</code> <code>x, y=y, x</code> <code>print x, y</code>	30 20
4 <code>K=[1,4,5,6]</code> <code>for a in K:</code> <code> print a*2</code>	2 8 10 12
5 <code>sxoleio=[]</code> <code>for a in range(1,5):</code> <code> sxoleio.append(a*(a+1))</code> <code>print sxoleio</code>	[2, 6, 12, 20]

2.2

Αριθμός γραμμής	Σφάλμα	Διόρθωση σφάλματος	Κατηγορία σφάλματος
1	<code>s = i = 1.0</code>	<code>s = i = 0.0</code>	Σημασιολογικό ή λογικό σφάλμα
3	<code>age >= 0</code>	<code>age > 0</code>	Σημασιολογικό ή λογικό σφάλμα
4	<code>s = s + i</code>	<code>s = s + age</code>	Σημασιολογικό ή λογικό σφάλμα
7	<code>if i != 0</code>	<code>if i != 0:</code>	Συντακτικό λάθος
8	<code>mo = age / i</code>	<code>mo = s / i</code>	Σημασιολογικό ή λογικό σφάλμα

ΘΕΜΑ 4

Η συνδρομητική υπηρεσία audio streaming SPOT AE χρεώνει τους συνδρομητές της για κάθε τραγούδι που κατεβάζουν, σύμφωνα με τον παρακάτω πίνακα:

Αριθμός τραγουδιών	Χρέωση (ευρώ / τραγούδι)
Μέχρι και 30	0.0020
Από 31 μέχρι και 60	0.0016
Άνω των 60	0.0010

Το μηνιαίο πάγιο είναι 3 ευρώ ενώ υπάρχει και ΦΠΑ 12% επί της συνολικής χρέωσης. Να αναπτύξετε πρόγραμμα σε Python το οποίο:

4.1 Να διαβάζει επαναληπτικά τον αριθμό των τραγουδιών που κατέβασε ένας συνδρομητής για κάθε ημέρα του μήνα. Απαιτείται έλεγχος εγκυρότητας δεδομένων για τον αριθμό των τραγουδιών ώστε να είναι θετικός αριθμός ή 0. Η διαδικασία να τερματίζεται όταν δοθεί η τιμή -1. Να θεωρήσετε ότι ο μήνας έχει 30 ημέρες.

Μονάδες 10

4.2 Να εμφανίζει στην οθόνη τη χρέωση του συνδρομητή για κάθε ημέρα και τη συνολική χρέωση για ολόκληρο το μήνα. Για το σκοπό αυτό γίνεται κλήση της συνάρτησης `xreosi()`, η οποία δέχεται σαν παράμετρο τον αριθμό των τραγουδιών και επιστρέφει τη χρέωση.

Μονάδες 10

4.3 Να εμφανίζει στην οθόνη το πλήθος των τραγουδιών που κατέβασε συνολικά ο συνδρομητής για ολόκληρο τον μήνα.

Μονάδες 5

Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1  # Δημιουργία συνάρτησης
2  def xreosi(t):
3      if t<30:
4          x=.0020*t
5      elif t<=60:
6          x=.0016*t
7      else:
8          x=.0010*t
9      return x
10 # Αρχικοποίηση μεταβλητών
11 pagio=3
12 s_plithos=0
13 xr=s_xr=0
14 # 4.1
15 for k in range (1, 31):
16     print "Ημέρα: ", k
17     pl_tr=int(input("Εισάγετε αριθμό τραγουδιών (θετικός ή 0): "))
18 # Έλεγχος εγκυρότητας δεδομένων
19     while pl_tr<0 and pl_tr!=-1:
20         pl_tr=int(input("Εισάγετε αριθμό τραγουδιών (θετικός ή 0): "))
21 # Κλήση συνάρτησης
22     xr=xreosi(pl_tr)
23 # 4.2
24     print "Η χρέωση της ημέρας ", k, " είναι: ", xr
25     s_xr+=xr
26 # 4.3 Συνολικό πλήθος τραγουδιών
27     s_plithos+=pl_tr
28 # 4.2 Συνολική χρέωση
29     print "Η συνολική χρέωση του συνδρομητή για τον μήνα είναι: ", (s_xr+pagio)*1.12, " ευρώ"
30 # 4.3
31     print "Το συνολικό πλήθος τραγουδιών είναι: ", s_plithos
```

ΘΕΜΑ 2

2.1 Να υλοποιήσετε τις παρακάτω προτάσεις στη γλώσσα προγραμματισμού Python:

1. Να γράψετε μια απλή γραμμή σχολίων με το κείμενο 'Hallo World!'.
2. Να οριστεί η μεταβλητή `number` η οποία να περιέχει την αριθμητική τιμή 2.135.
3. Να οριστεί η μεταβλητή `number_tetragwono` η οποία να περιέχει το τετράγωνο της μεταβλητής `number`.
4. Να εμφανιστούν στην οθόνη οι τιμές των μεταβλητών `number`, `number_tetragwono`.
5. Να οριστεί η μεταβλητή `bool` που να περιέχει τη λογική τιμή `True`.
6. Να οριστεί η μεταβλητή `seed` που να περιέχει την ακέραια τιμή 10.
7. Να οριστεί η μεταβλητή `x` που να περιέχει το κείμενο `Python`.

Μονάδες 14

2.2 Να αντιστοιχηθούν οι τιμές της στήλης A με τους τύπους δεδομένων της στήλης B.

Στήλη A (Τιμή)	Στήλη B (Τύπος δεδομένων)
1. -16	A. int (ακέραια) B. float (κινητής υποδιαστολής) Γ. str (συμβολοσειρά) Δ. bool (λογική)
2. 26.9	
3. "True"	
4. False	
5. "124.32"	
6. "Hallo world"	
7. 45 / 6	
8. 30 % 4	
9. True	
10. 25	
11. 25.0	

Μονάδες 11

ΘΕΜΑ 2

2.1

1. `# "Hallo World!"`
2. `number = 2.135`
3. `number_tetragvno = number**2`
4. `print number, number_tetragvno`
5. `bool = True`
6. `seed = 10`
7. `x = "Python"`

2.2

1A, 2B, 3Γ, 4Δ, 5Γ, 6Γ, 7B, 8A, 9Δ, 10A, 11B

ΘΕΜΑ 2

2.1 Στο μοντέλο της σπείρας, η ανάπτυξη ακολουθεί μια εξελικτική διαδικασία με την επαναληπτική εκτέλεση ενός κύκλου φάσεων, οι οποίες εμφανίζονται αριθμημένες, αλλά όχι με τη σωστή σειρά:

1. Συγγραφή κώδικα
2. Τεκμηρίωση
3. Σχεδίαση
4. Ανάλυση του προβλήματος
5. Έλεγχος και εκσφαλμάτωση

Να γράψετε στο γραπτό σας τη σωστή σειρά των φάσεων.

Μονάδες 5

2.2 Να επιλέξετε τις σωστές απαντήσεις (περισσότερες από μία):

Η ευχρηστία του συστήματος επιτυγχάνεται, όταν η επικοινωνία ενός χρήστη με ένα σύστημα υπολογιστή γίνεται μέσω ενός περιβάλλοντος που χαρακτηρίζεται από:

1. ευκολία εκμάθησης
2. υψηλή απόδοση εκτέλεσης έργου
3. χαμηλή συχνότητα εμφάνισης λαθών χρήστη
4. λειτουργία μηχανής αναζήτησης
5. ευκολία συγκράτησης της γνώσης της χρήσης του
6. υποκειμενική ικανοποίηση του χρήστη

Μονάδες 5

2.3

Δίνεται ο παρακάτω κώδικας σε Python:

```

1 i=0
2 a=5
3 while(i <= a):
4     if (i % 2 == 0 or a % 3 == 0):
5         print "Υπέροχα", "!"
6     else:
7         print ("i", "a")
8     i += 1
9     a -= 1

```

Επίσης, δίνεται το ακόλουθο υπόδειγμα πίνακα τιμών:

Αριθμός εντολής	i	a	Οθόνη
1	0		
...	...	...	...

Να συμπληρώσετε στο γραπτό σας τον πίνακα τιμών, δημιουργώντας μια νέα γραμμή στον πίνακα για κάθε εντολή εκχώρησης που εκτελείται και μια νέα γραμμή για κάθε εντολή εξόδου (Οθόνη) που εκτελείται.

Μονάδες 15

ΘΕΜΑ 2

2.1

4 – 3 – 1 – 5 – 2

2.2

1 – 2 – 3 – 5 – 6

2.3

Αριθμός εντολής	i	a	Οθόνη
1	0		
2		5	
5			Υπέροχα !
8	1		
9		4	
7			(1 , 4)
8	2		
9		3	
5			Υπέροχα !
8	3		
9		2	

ΘΕΜΑ 4

Ένας ανελκυστήρας έχει μέγιστο όριο ασφαλείας τα 650 κιλά. Να αναπτύξετε πρόγραμμα στη γλώσσα προγραμματισμού Python το οποίο:

4.1 Να διαβάζει το βάρος κάθε ατόμου (σε κιλά) που εισέρχεται στον ανελκυστήρα. Να γίνεται έλεγχος εγκυρότητας για το βάρος (να είναι γνησίως θετικό).

Μονάδες 6

4.2 Το πρόγραμμα θα τερματίζει όταν ο ανελκυστήρας γεμίσει σε σχέση με το μέγιστο επιτρεπτό όριο ασφαλείας βάρους, δηλαδή όταν προσπαθήσει να εισέλθει άτομο με βάρος που αθροιστικά με τα προηγούμενα άτομα ξεπερνούν το κατώφλι των 650 κιλών.

Μονάδες 13

4.3 Στη συνέχεια, να εμφανίζει το πλήθος και το συνολικό βάρος των ατόμων που κατάφεραν να εισέλθουν στον ανελκυστήρα.

Μονάδες 6

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1  # Αρχικοποιήσεις μεταβλητών
2  persons = 0
3  capacity = 650
4  sum = 0
5  # Εισαγωγή δεδομένων
6  weight = int(raw_input("Βάρος ατόμου σε κιλά = "))
7  while weight <= 0 : # Έλεγχος εγκυρότητας δεδομένων
8      weight = int(raw_input("Βάρος ατόμου σε κιλά = "))
9  # Βρόχος επανάληψης για έλεγχο εισερχομένων ατόμων
10 while weight <= capacity:
11     capacity = capacity - weight
12     persons = persons + 1
13     sum = sum + weight
14     weight = int( raw_input("Βάρος ατόμου σε κιλά = "))
15     while weight <=0 :
16         weight = int( raw_input("Βάρος ατόμου σε κιλά = "))
17 # Εμφάνιση αποτελεσμάτων
18 print "Πλήθος ατόμων " , persons
19 print "Συνολικό βάρος εισερχομένων ατόμων " , sum
```

ΘΕΜΑ 2

2.1 Να γράψετε τις τιμές που εισάγονται στη λίστα σε κάθε περίπτωση σαν αποτέλεσμα της εκτέλεσης των παρακάτω τμημάτων κώδικα σε γλώσσα προγραμματισμού Python:

α)

```
sq = []  
for x in range (1,11,2):  
    sq.append(x**2)
```

β)

```
sq = []  
for x in range (2,8,2):  
    sq.append(x**2)
```

Μονάδες 12

2.2 Να συμπληρώσετε τα τέσσερα (4) αριθμημένα κενά στο παρακάτω πρόγραμμα σε γλώσσα προγραμματισμού Python έτσι ώστε να εμφανίζει στην οθόνη τον πίνακα της προπαίδειας ενός αριθμού που θα δίνεται από τον χρήστη.

```
num = int( (1) ("Δώσε έναν αριθμό: "))  
for i in (2) ( (3) ,11):  
    print( (4) ,"*",num,"=",i*num)
```

Μονάδες 13

ΘΕΜΑ 2

2.1

α) Τα τετράγωνα των μονών αριθμών από το 1 μέχρι και το 9.

β) Τα τετράγωνα των άρτιων αριθμών από το 2 μέχρι και το 6.

γ) Τους ακέραιους αριθμούς από το 3 μέχρι και το 11.

2.2

1 – input

2 – range

3 – 1

4 – i

ΘΕΜΑ 4

Στον διαγωνισμό Πληροφορικής Bebras συμμετέχουν δέκα (10) μαθητές. Κάθε μαθητής που συμμετέχει λαμβάνει μια τελική βαθμολογία από 1 έως και 100 ακέραιες μονάδες.

Να γράψετε πρόγραμμα σε γλώσσα προγραμματισμού Python, το οποίο:

4.1 Να διαβάζει το ονοματεπώνυμο κάθε μαθητή και τη βαθμολογία που έλαβε. Τα στοιχεία αυτά καταχωρίζονται σε δύο λίστες. Να μην γίνει έλεγχος ορθότητας εισαγωγής δεδομένων.

Μονάδες 5

4.2 Να υπολογίζει και να εμφανίζει τον μέσο όρο της βαθμολογίας όλων των μαθητών.

Μονάδες 10

4.3 Να εντοπίζει και να εμφανίζει στην οθόνη τα ονοματεπώνυμα και τη βαθμολογία όλων των μαθητών των οποίων η τελική βαθμολογία είναι μεγαλύτερη ή ίση από τον μέσο όρο.

Μονάδες 10

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1 VATHMOS=[]
2 NAME=[]
3 for i in range(10):
4     n=raw_input("δώσε όνομα ")
5     NAME.append(n)
6     v=int(input("Δώσε βαθμό: "))
7     VATHMOS.append(v)
8 # ερώτημα 4.2
9 s=0.0
10 for number in VATHMOS :
11     s= s + number
12 mo=s / 50
13 print "Μέσος όρος βαθμολογίας μαθητών: ", mo
14 # ερώτημα 4.3
15 for i in range(50) :
16     if VATHMOS[i] >= mo :
17         print "όνομα μαθητή: ", NAME[i], "Βαθμός: ", VATHMOS[i]
```

ΘΕΜΑ 2

2.1 Να γράψετε στο γραπτό σας το αποτέλεσμα που εμφανίζεται στην οθόνη μετά την εκτέλεση καθεμιάς από τις παρακάτω εντολές σε γλώσσα προγραμματισμού Python:

- α. `range(3,9)`
- β. `range(2,13,3)`
- γ. `a = "two"`
 `print a*2`
- δ. `x=4`
 `y=5`
 `print 2*x-y`
- ε. `a = 3`
 `print a**2`

Μονάδες 10

2.2 Να γράψετε στο γραπτό σας ότι εμφανίζεται στην οθόνη κατά την εκτέλεση του παρακάτω προγράμματος σε Python:

```
a=2
x=1
b=0
while x<3:
    b=b+1
    a=a*x
    print "a=", a, "x=", x
    if b%2 == 0:
        x=x+2
print "b=", b
```

Μονάδες 15

ΘΕΜΑ 2

2.1

α . [3, 4, 5, 6, 7, 8]

β . [2, 5, 8, 11]

γ . twotwo

δ . 3

ϵ . 9

2.2

$a = 2 \ x = 1$

$a = 2 \ x = 1$

$b = 2$

ΘΕΜΑ 4

Να γράψετε πρόγραμμα σε γλώσσα προγραμματισμού Python το οποίο να υλοποιεί τα παρακάτω:

4.1. Να διαβάζει επαναληπτικά τον τίτλο κινηματογραφικής ταινίας και το πλήθος των θεατών που την παρακολούθησαν. Το πλήθος των θεατών αποθηκεύεται σε μια λίστα με όνομα PLITHOS. Η εισαγωγή των στοιχείων θα τερματίζεται όταν δοθεί ως τίτλος ταινίας η λέξη «END».

Μονάδες 10

4.2. Να εντοπίζει και να εμφανίζει το μεγαλύτερο πλήθος θεατών στην πιο δημοφιλή ταινία. Να θεωρήσετε ότι η ταινία με το μεγαλύτερο πλήθος θεατών είναι μοναδική.

Μονάδες 10

4.3. Να υπολογίζει και εμφανίζει στην οθόνη τον μέσο όρο των θεατών όλων των ταινιών.

Μονάδες 5

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1  pl = 0 # Αρχικοποιήσεις
2  s = 0
3  PLITHOS=[ ]
4  n=raw_input("Δώσε τίτλο κινηματογραφικής ταινίας: ") # 4.1
5  while n != "END" : # Βρόχος επανάληψης
6      pl=pl+1
7      p=int(input("Δώσε αριθμό θεατών: "))
8      PLITHOS.append(p)
9      n=raw_input("Δώσε τίτλο κινηματογραφικής ταινίας: ")
10 # 4.2
11 max=PLITHOS[0]
12 for number in PLITHOS :
13     if number > max :
14         max=number
15 print "Το μέγιστο πλήθος θεατών σε μια ταινία είναι: ", max
16 # 4.3
17 for i in range(pl) :
18     s=s+PLITHOS[i]
19 mo=s/pl
20 print "Ο μέσος όρος των θεατών είναι: ", mo
```

ΘΕΜΑ 2

2.1. Να γράψετε στο γραπτό σας τι εμφανίζει το παρακάτω πρόγραμμα κατά την εκτέλεση του.

```
1  def comput(y,x):  
2      y=x+4  
3      x=y/2  
4      print x,y  
5      return x-y  
6  
7  x=5  
8  y=2  
9  print y**2,x  
10 z=comput(x,y)  
11 x=y+z  
12 y=x+z  
13 print z,x
```

Μονάδες 12

2.2. Η βαθμολογική κλίμακα, με βάση την οποία υπολογίζονται οι βαθμοί επίδοσης των μαθητών σε όλα τα μαθήματα, κυμαίνεται από μηδέν (0) έως είκοσι (20) και προσδιορίζεται ως εξής:

Βαθμολογική Κλίμακα	Χαρακτηρισμός
0 - 5	Κακώς
5,1 - 9,9	Ανεπαρκώς
10 - 13	Σχεδόν καλώς
13,1 - 16	Καλώς
16,1 - 18	Λίαν καλώς
18,1 - 20	Άριστα

Να γράψετε τμήμα προγράμματος το οποίο θα εμφανίζει τον κατάλληλο χαρακτηρισμό επίδοσης ανάλογα με την βαθμολογική κλίμακα.

Μονάδες 13

ΘΕΜΑ 2

2.1. Οι τιμές που θα εμφανιστούν είναι:

4 5
3 6
-3 -1

2.2.

```
1  if B>0 and B<=5:  
2      print 'Κακώς'  
3  elif B<=9.9:  
4      print 'Ανεπαρκώς'  
5  elif B<=13:  
6      print 'Σχεσών Καλώς'  
7  elif B<=16:  
8      print 'Καλώς'  
9  elif B<=18:  
10     print 'Λίαν Καλώς'  
11 else:  
12     print 'Άριστα'
```

ΘΕΜΑ 4

Να δημιουργηθεί πρόγραμμα σε Python το οποίο θα κάνει προσομοίωση ενός ψηφιακού χρονομέτρου ως εξής:

4.1. Να διαβάζει μια ακέραια θετική τιμή μεταξύ του 3 και του 15, η οποία θα συμβολίζει το χρόνο σε λεπτά που θα προσομοιώνει το ψηφιακό χρονόμετρο. Να πραγματοποιηθεί έλεγχος εγκυρότητας τιμής.

Μονάδες 8

4.2. Να εμφανίζει στην οθόνη το χρόνο μειωμένο κατά 1 δευτερόλεπτο, συνεχόμενα και κατά φθίνουσα σειρά μέχρι την ένδειξη 0 min:00sec. Για παράδειγμα (για είσοδο της τιμής 3 να εμφανίζει διαδοχικά τα μηνύματα σε διαφορετικές γραμμές :

Ο χρόνος θα λήξει σε:

3 min:00 sec

2 min:59 sec

2 min:58 sec

...

1 min:15 sec

...

0 min:01 sec

0 min:00 sec

Μονάδες 15

4.3. Μόλις μηδενιστεί ο χρόνος εμφανίζει το μήνυμα «Τέλος χρόνου».

Μονάδες 2

ΘΕΜΑ 4

```
1  time=int(input('Δώσε τον χρόνο που θέλεις να ξεκινήσει ο αγώνας [2min-5min]')) #4.1
2  # 4.1 Έλεγχος αν η τιμή να είναι από 2 έως 5 min
3  while time<2 or time>5:
4      print "Έδωσες λάθος χρόνο"
5      time=int(input('Δώσε χρόνο [από 2min έως 5min]'))
6
7  print 'Ο αγώνας θα ξεκινήσει σε:'
8  syn_sec=time*60          #4.2 τα συνολικά sec που αντιστοιχούν στο χρόνο εισόδου
9  while syn_sec>0:        #4.2 μέχρι να μηδενιστεί ο χρόνος
10     min=syn_sec/60        #4.2 Υπολογίζεται ο χρόνος που απομένει σε λεπτά
11     sec=syn_sec%60        #4.2 Τα δευτερόλεπτα που απομένουν
12     print min,' min:',sec,'sec' #4.2 εμφάνιση χρόνου σε min και sec
13     syn_sec=syn_sec-1    #4.2 μείωση του χρόνου κατά ένα sec
14 print 'Καλή επιτυχία'    #4.3
```

ΘΕΜΑ 2

2.2. Να γράψετε στο γραπτό σας τι εμφανίζει το παρακάτω πρόγραμμα κατά την εκτέλεση του.

```
1  x="ΤΕΛΟΣ ΜΑΘΗΜΑΤΩΝ"  
2  print x[:10]  
3  print x[6:len(x)]  
4  print "για τους ".upper() + x[6:10]+x[0:2]+x[4]  
5  print x.find('A')  
6  print x[8:10]+x[12]+x[7]  
7  print len(x[3:9])
```

Μονάδες 13

2.2. Δίνεται το παρακάτω τμήμα προγράμματος σε γλώσσα προγραμματισμού Python:

```
1  L=[_(1)_]  
2  for i in range(_(2)_,_(3)_,_(4)_):  
3      L.append(_(5)_)  
4  print _(6)_
```

Στο τμήμα προγράμματος υπάρχουν υπογραμμισμένα κενά τα οποία έχουν αριθμηθεί. Να γράψετε στο τετράδιό σας τους αριθμούς 1, 2, 3, 4, 5 και 6 που αντιστοιχούν στα κενά του παραπάνω τμήματος προγράμματος και δίπλα αυτό που πρέπει να συμπληρωθεί, ώστε να δημιουργηθεί η παρακάτω λίστα:

$L = [80, 72, 64, 56, 48, 40, 32, 24, 16, 8]$

Μονάδες 12

2.1

ΤΕΛΟΣ

ΜΑΘΗΜΑΤΩΝ

ΓΙΑ ΤΟΥΣ ΜΑΘΗΤΕΣ

7

ΘΗΤΑ

6

2.2

(1) []

(2) 80

(3) 7

(4) -8

(5) [i]

(6) L

ΘΕΜΑ 4

Ένας ιδιοκτήτης μιας μονοκατοικίας επένδυσε κάποια χρήματα για να εγκαταστήσει κάποια φωτοβολταϊκά πάνελ στην ταράτσα του σπιτιού του με σκοπό να αυτονομηθεί ενεργειακά.

Να φτιάξετε πρόγραμμα σε Python το οποίο:

4.1. Να διαβάξει την μηνιαία κατανάλωση που είχε η οικία του και την μηνιαία παραγωγή που απέδωσε το φωτοβολταϊκό, σε κιλοβατώρες το προηγούμενο έτος, καταχωρίζοντας τις τιμές αυτές στις λίστες KAT και PAR αντίστοιχα. Θεωρήστε ότι οι τιμές είναι θετικές.

Μονάδες 8

4.2. Να υπολογίζει και εμφανίζει πόση ήταν η ετήσια κατανάλωση της οικίας του και η ετήσια παραγωγή του φωτοβολταϊκού, μετά από κατάλληλα διαμορφωμένα μηνύματα, και αν απέδωσε το κόστος της επένδυσης.

Μονάδες 10

4.3. Να υπολογίζει και εμφανίζει πόσους μήνες αυτονομήθηκε ενεργειακά η οικία δηλαδή το ρεύμα που παρήγαγε το φωτοβολταϊκό του ήταν μεγαλύτερο από την απαιτούμενη κατανάλωση της οικίας του.

Μονάδες 7

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
    print b
```

ΘΕΜΑ 4

```
1      #Αρχικοποίηση λιστών
2      KAT=[]
3      PAR=[]
4      #4.1 Εισαγωγή δεδομένων και καταχώρηση σε λίστα
5      for i in range(12):
6          k=float(input('Δώσε την κατανάλωση του '+str(i+1)+'ου μήνα'))
7          p=float(input('Δώσε την παραγωγή του '+str(i+1)+'ου μήνα'))
8          KAT=KAT+[k]
9          PAR.append(p)
10     #4.2 Υπολογισμός ετήσιας κατανάλωσης και ετήσιας παραγωγής
11     sk=sp=0.0
12     for i in range(12):
13         sk=sk+KAT[i]
14         sp=sp+PAR[i]
15     #Έλεγχος ποια κατανάλωση ήταν μεγαλύτερη
16     print 'Η ετήσια κατανάλωση σε κιλοβατώρες είναι',sk
17     print 'Η ετήσια παραγωγή σε κιλοβατώρες του φωτοβολταϊκού είναι',sp
18     if sk>sp:
19         print "Η κατανάλωση ήταν μεγαλύτερη από",
20         print "την παραγωγή και δεν αυτονομήθηκε η οικία ενεργειακά"
21     elif sk<sp:
22         print " Η παραγωγή ήταν μεγαλύτερη από την κατανάλωση ",
23         print " και αυτονομήθηκε η οικία ενεργειακά. ΑΞΙΖΕ Η ΕΠΕΝΔΥΣΗ"
24     else:
25         print " Η παραγωγή ήταν ίση με την κατανάλωση ",
26         print " και αυτονομήθηκε η οικία ενεργειακά. ΑΞΙΖΕ Η ΕΠΕΝΔΥΣΗ"
27     #4.3 Πόσους μήνες υπήρχε ενεργειακή αυτονομία
28     # παραγόμενες κιλοβατώρες>καταναλώμενες κιλοβατώρες
29     pl=0
30     for i in range(12):
31         if PAR[i]>=KAT[i]:
32             pl=pl+1
33     print "Η οικία αυτονομήθηκε ενεργειακά",pl," μήνες"
```

ΘΕΜΑ 2

2.1. Το ακόλουθο πρόγραμμα σε Python έχει γραφεί να δέχεται μια συμβολοσειρά και να υπολογίζει το μήκος της. Στη συνέχεια αντιστρέφει τη συμβολοσειρά και την εμφανίζει ανεστραμμένη. Στο πρόγραμμα, όμως, υπάρχουν πέντε (5) λάθη. Για κάθε ένα από τα λάθη να γράψετε στο γραπτό σας τον αριθμό της γραμμής στην οποία το εντοπίσατε, να περιγράψετε το λάθος και να χαρακτηρίσετε το λάθος ως συντακτικό, λογικό ή χρόνου εκτέλεσης. Δεν υπάρχουν περισσότερα του ενός λάθη στην ίδια γραμμή.

```
1  x='Python'
2  pl=0
3  for i in x:
4      pl=1
5  print "Το μήκος της συμβολοσειράς είναι"pl
6  αντιστροφή λέξης
7  new_x=""
8  for i in x:
9      new_x=i+new_x
10 for i in range(len(new_x)+2 ):
11     print new_x[i],
```

Μονάδες 15

2.2. Να γράψετε στο γραπτό σας τους αριθμούς 1, 2, 3, 4, 5, από τη **Στήλη Α** του παρακάτω πίνακα και δίπλα ένα από τα γράμματα **α, β, γ, δ, ε, στ** της **Στήλης Β**, που δίνει τη σωστή αντιστοίχιση. Σημειώνεται ότι ένα γράμμα από τη **Στήλη Β** θα περισσέψει.

Στήλη Α Εντολή	Στήλη Β Αλγοριθμικές Δομές
1. X=10	α. Δομή Επανάληψης
2. if X>=10:	β. Εντολή Εισόδου
3. while x!=0:	γ. Εντολή εκχώρησης τιμής
4. print x	δ. Δομή Επιλογής
5. x=input('Give a price')	ε. Συνάρτηση
	στ. Εντολή Εξόδου

Μονάδες 10

2.1

Αριθμός Γραμμής	Είδος Σφάλματος	Περιγραφή και πρόταση διόρθωσης
1.	Συντακτικό	λείπουν τα εισαγωγικά στο τέλος της συμβολοσειράς. Το σωστό θα είναι: <code>x='Python'</code>
4.	Λογικό	Η μεταβλητή <code>pl</code> θα πρέπει να αυξάνεται κατά 1 και όχι να παίρνει συνέχεια την τιμή 1. Δεν μετράει σωστά το πλήθος των γραμμών της συμβολοσειράς. Το σωστό θα είναι: <code>pl=pl+1</code> .
5.	Συντακτικό	Μετά την συμβολοσειρά πρέπει να υπάρχει κόμμα στην τιμή της μεταβλητής που θέλουμε να εμφανιστεί. Το σωστό είναι: <code>print "Το μήκος της συμβολοσειράς είναι",pl</code>
6.	Συντακτικό	Λείπει το σύμβολο <code>#</code> που δηλώνει ότι η γραμμή αυτή είναι σχόλιο. Το σωστό είναι: <code>#αντιστροφή λέξης</code>
11.	Χρόνου εκτέλεσης	Γίνεται αναφορά σε στοιχείο το οποίο είναι εκτός συμβολοσειράς στην εντολή <code>print new_x[i]</code> , . Το σωστό θα είναι: <code>for i in range(len(new_x)):</code>

2.2

1. γ
2. δ
3. α
4. στ
5. β

ΘΕΜΑ 4

Μια εταιρεία, θέλοντας να ενισχύσει οικονομικά τους εργαζόμενους της, την περίοδο των εορτών του Πάσχα αποφάσισε να τους παρέχει ένα πρόσθετο οικονομικό βοήθημα (επίδομα) σύμφωνα με τον παρακάτω πίνακα:

Μισθός	Επίδομα
Μικρότερος από 800€	200€
801€ - 1200€	150€
1201€ - 1700€	100€
>1700€	70€

Να γράψετε πρόγραμμα σε Python το οποίο:

4.1. Να διαβάζει το όνομα και τον μισθό του εργαζόμενου της εταιρείας. Να γίνεται έλεγχος εγκυρότητας τιμών ώστε ο μισθός να είναι μεγαλύτερος των 450€ και μικρότερος των 2000€. Η είσοδος των δεδομένων θα επαναλαμβάνεται μέχρι να δοθεί ως όνομα το κενό (' ').

Μονάδες 7

4.2. Θα καλεί την συνάρτηση `dwro` η οποία θα δέχεται τον μισθό του υπαλλήλου και θα επιστρέφει το ποσό του επιδόματος που δικαιούται, σύμφωνα με τον παραπάνω πίνακα. Στη συνέχεια το πρόγραμμα θα εμφανίζει το όνομα του υπαλλήλου και το ποσό του επιδόματος που θα λάβει ως οικονομική ενίσχυση.

Μονάδες 10

4.3. Να εμφανίζει το όνομα του τελευταίου υπαλλήλου που πήρε το μεγαλύτερο επίδομα.

Μονάδες 8

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1      #4.2. Συνάρτηση υπολογισμού επιδόματος
2      def dwro(poso):
3          if poso < 800:
4              epidoma=200
5          elif poso<=1200:
6              epidoma=150
7          elif poso<=1700:
8              epidoma=100
9          else:
10             epidoma=70
11
12         return epidoma  #το ποσο της οικονομικής ενίσχυσης (επίδομα)
13
14     #Κυρίως Πρόγραμμα
15     #Αριχοποίηση μεταβλητών
16     pl100=0  #πλήθος που πήραν επίδομα >απο 100€
17     pl=0      #πλήθος εργαζομένων
18     meg_epidoma=-1 #μεγαλύτερο επίδομα
19     #4.1 Εισαγωγή δεδομένων
20     name=raw_input('Δώσε το όνομα του υπαλλήλου')
21     while name!=' ':      #Έλεγχος μέχρι να διαβάσει το κενό
22         #4.1 Εισαγωγή του μισθού του υπαλλήλου
23         salary=float(input('Δώσε τον μισθό του '+name))
24         while salary<450 or salary>2000:  #έλεγχος αποδεκτών τιμών
25             salary=float(input('Δώσε ένα μισθό μεταξύ 450 και 2000'))
26
27         pl=pl+1      #πλήθος εργαζομένων
28         extra=dwro(salary)  #κλίση συνάρτησης για υπολογισμό επιδόματος
29         print "Ο υπάλληλος ",name," θα πάρει οικονομική ενίσχυση ",extra," €" #4.2
30         #4.3 Υπολογισμός του τελευταίου που πήρε το μεγαλύτερο επίδομα
31         if extra>=meg_epidoma:
32             meg_epidoma=extra
33             on_ypal=name
34         print "Ο ",on_ypal,"πήρε το μεγαλύτερο επίδομα το οποίο είναι ",meg_epidoma," €" #4.3
```

ΘΕΜΑ 2

2.1. Να γράψετε στο γραπτό σας τους αριθμούς 1, 2, 3, 4, 5, από τη **Στήλη Α** του παρακάτω πίνακα και δίπλα ένα από τα γράμματα **α, β, γ, δ, ε, στ** της **Στήλης Β**, που δίνει τη σωστή αντιστοίχιση. Σημειώνεται ότι ένα γράμμα από τη **Στήλη Β** θα περισσέψει.

Στήλη Α	Στήλη Β
1. $X \% 3 == 0$:	α. Τελευταίο Ψηφίο αριθμού
2. $X \% 2 == 1$:	β. Πρώτο ψηφίο ενός τριψήφιου αριθμού
3. $X \% 10$	γ. Ζυγός αριθμός
4. $X \% 100$	δ. Πολλαπλάσιο του 3
5. $(X \% 100) / 10$	ε. Μονός αριθμός
	στ. Μεσαίο ψηφίο ενός τριψήφιου αριθμού

Μονάδες 10

2.2. Δίνεται το παρακάτω τμήμα προγράμματος σε γλώσσα προγραμματισμού Python

```
1 s='Python'
2 for i in len(s):
3     print s[i]
4 print s+' 2.7.10'
```

Επίσης δίνεται το παρακάτω υπόδειγμα πίνακα:

	i	Έξοδος
Αρχικές Τιμές		
1η επανάληψη	...	...
...	...	...

Να μεταφέρετε στο γραπτό σας τον παραπάνω πίνακα και να συμπληρώσετε τις τιμές του i και τι θα εμφανίζεται σε κάθε επανάληψη, προσθέτοντας μια νέα γραμμή στον πίνακα για κάθε επανάληψη του προγράμματος.

Μονάδες 15

2.1

1. δ
2. ε
3. α
4. β
5. στ

2.2

	i	Έξοδος
Αρχικές Τιμές		
1η επανάληψη	0	p
2η επανάληψη	1	y
3η επανάληψη	2	t
4η επανάληψη	3	h
5η επανάληψη	4	o
6η επανάληψη	5	n
		Python 2.7.10

ΘΕΜΑ 4

Ένας αθλητής θέλει να καταγράψει πόσες θερμίδες καταναλώνει καθημερινά για να έχει μια ισορροπημένη διατροφή. Να γράψετε πρόγραμμα σε Python το οποίο :

4.1. Να διαβάζει επαναληπτικά το όνομα της τροφής και τον αριθμό των θερμίδων που έχει αυτή, ελέγχοντας η τιμή να είναι θετική και μεγαλύτερη του 0 (έλεγχος εγκυρότητας). Η εισαγωγή των στοιχείων θα τερματίζεται όταν δοθεί ως όνομα τροφής η λέξη "ΤΕΛΟΣ".

Μονάδες 8

4.2. Να υπολογίζει το σύνολο των θερμίδων που κατανάλωσε στο τέλος της ημέρας και να εμφανίζεται μετά το μήνυμα «Οι συνολικές θερμίδες που κατανάλωσες είναι»

Μονάδες 6

4.3 Να υπολογίζει την τροφή με τις περισσότερες θερμίδες που κατανάλωσε και να εμφανίζει το όνομα της τροφής αυτής και τον αριθμό των θερμίδων που είχε αυτή.

Μονάδες 7

4.4. Να υπολογίζει και εμφανίζει το πλήθος των τροφών που κατανάλωσε και είχαν πάνω από 500 θερμίδες.

Μονάδες 4

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1  #αρχικοποίηση μεταβλητών
2  pl=0      #πλήθος τροφών πάνω από 500 θερμίδες
3  syn_cal=0  #σύνολο θερμίδων που κατανάλωσε
4  max_cal=-1 # μέγιστη τιμή
5  #4.1. Εισαγωγή δεδομένων
6  on=raw_input('Δώσε ονομασία τροφής που θέλεις να καταναλώσεις')
7  #Όσο όνομα τροφής είναι διαφορετικό του TELOS
8  while on!='TELOS':
9      #4.1. Εισαγωγή θερμίδων τροφής
10     cal=int(input('Δώσε των αριθμό των θερμίδων της '+on+' τροφής'))
11     #Έλεγχος αποδεκτών τιμών για θετική τιμή θερμίδων
12     while cal<=0:
13         cal=int(input('Δώσε θετικό αριθμό θερμίδων για την '+on+' τροφή'))
14     #4.2. Συνολικός αριθμός θερμίδων που καταναλώθηκαν
15     syn_cal=syn_cal+cal
16     #4.3. Υπολογισμός μέγιστης κατανάλωσης θερμίδων
17     if cal>max_cal:
18         max_cal=cal
19         name_cal=on
20     #4.4. πλήθος τροφών με περισσότερες από 500 θερμίδες
21     if cal>500:
22         pl=pl+1
23     on=raw_input('Δώσε επόμενο όνομα τροφής ή TELOS')
24 #Εκτύπωση ζητούμενων
25 print 'Οι συνολικές θερμίδες που κατανάλωσες είναι ',syn_cal  #4.1
26 print 'Η τροφή με τις περισσότερες θερμίδες είναι η ',name_cal,' με',max_cal,' θερμίδες'  #4.2
27 print pl,'από τις τροφές που κατανάλωσες ήταν πάνω από 500 θερμίδες'  #4.3
```

ΘΕΜΑ 4

Οι μαθητές της Γ' Λυκείου για να ενισχυθούν οικονομικά για την πολυήμερη εκδρομή τους αποφάσισαν να κάνουν μια θεατρική παράσταση στην αίθουσα εκδηλώσεων του σχολείου τους με την ενδεικτική τιμή των 2€. Η χωρητικότητα της αίθουσας εκδηλώσεων είναι 200 άτομα. Να γράψετε πρόγραμμα σε Python το οποίο:

4.1. Να διαβάζει το ποσό που δίνει κάθε θεατής για να παρακολουθήσει την θεατρική παράσταση μέχρι να δοθεί μια αρνητική τιμή ή μέχρι να καλυφθούν όλες οι θέσεις της αίθουσας. Σε περίπτωση που καλύφθηκαν όλες οι θέσεις να εμφανίζει το μήνυμα “Συγχαρητήρια η αίθουσα γέμισε” διαφορετικά πόσες θέσεις έμειναν κενές.

Μονάδες 7

Στη συνέχεια το πρόγραμμα να υπολογίζει και εμφανίζει μετά από κατάλληλα μηνύματα:

4.2. Τα συνολικά χρήματα που συγκεντρώθηκαν και το πλήθος των ατόμων που παρακολούθησαν την παράσταση.

Μονάδες 8

4.3. Το μεγαλύτερο ποσό που δόθηκε από τους θεατές που παρακολούθησαν την παράσταση.

Μονάδες 5

4.4. Το ποσοστό των θεατών που έδωσαν την ενδεικτική τιμή των 2€.

Μονάδες 5

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
    print b
```

ΘΕΜΑ 4

```
1  atoma=0      #atoma=τα άτομα που θα παρακολουθήσουν την παράσταση 4.2 ,
2  pl2=0        #pl2=πόσοι θα δώσουν πάνω από 2€ 4.4
3  synolo=0      # Το συνολικό ποσό που θα συγκεντρωθεί 4.2
4  poso=float(input('Δώσε το ποσό που θέλεις για να παρακολουθήσεις την παράσταση')) # 4.1
5  meg_poso=-1000 # 4.3 Αρχικοποίηση μεγίστου ποσού
6  while poso>0 and atoma<200: # Μέχρι να δωθεί αρνητικό ποσό ή μέχρι να γεμίσει η αίθουσα
7      atoma=atoma+1          #4.2
8      synolo=synolo+poso     #4.2
9      if poso>meg_poso:      #4.3
10         meg_poso=poso
11         if poso>2:          #4.4
12             pl2=pl2+1
13         poso=float(input('Δώσε επόμενο ποσό ή αρνητική τιμή για τέλος')) #4.1
14 if atoma==200:              #4.1 Έλεγχος αν γέμισε η αίθουσα εκδηλώσεων.
15     print 'Συγχαρητήρια η αίθουσα γέμισε'
16 else:
17     print 'Εμειναν κενές',200-atoma , ' θέσεις'
18 print 'Το συνολικό ποσό που συγκεντρώθηκε είναι ',synolo,' €'      #4.2
19 print 'Την παράσταση παρακολούθησαν ',atoma,' άτομα'              #4.2
20 if meg_poso!=-1000:         #4.3
21     print 'Το μεγαλύτερο ποσό που δόθηκε στην παράσταση είναι ',meg_poso,' €'
22 if atoma==0 or pl2==0:      #4.4
23     print 'Δεν διαβάστηκαν στοιχεία ή δεν υπήρξαν άτομα που έδωσαν ποσό > από 2€' #4.1,
24 else:
25     print 'Το ποσοστό των θεατών που έδωσαν ποσό > από 2 € είναι ', float(pl2)/atoma*100 , '%'
```

ΘΕΜΑ 2

2.1. Να γράψετε στο γραπτό σας τον αριθμό της πρότασης και δίπλα τις αντίστοιχες εντολές στο περιβάλλον IDLE της Python, που επιτελούν τις παρακάτω λειτουργίες:

1. Να εκχωρήσετε στη μεταβλητή `ar` την τιμή κινητής υποδιαστολής 12.40 .
2. Να αυξήσετε την μεταβλητής `ar` κατά 40% .
3. Να καταχωρήσετε στην μεταβλητή `new` το ακέραιο μέρος της μεταβλητής `ar`.
4. Στη μεταβλητή `first` να καταχωρίσετε το πρώτο ψηφίο του αριθμού 85.
5. Το τελευταίο ψηφίο του αριθμού 85 να καταχωρηθεί στη μεταβλητή `last`.
6. Να εκτυπώσετε το τετράγωνο της μεταβλητής `last`.
7. Να εκτυπώσετε την μέση τιμή των μεταβλητών `first` και `last`.

Μονάδες 7

2.2. Να γράψετε στο γραπτό σας το παρακάτω τμήμα προγράμματος χρησιμοποιώντας την εντολή επανάληψης **while** αντί της εντολής επανάληψης **for**, ώστε να εμφανίζονται τα ίδια αποτελέσματα.

```
1      for i in range (100,0,-2):  
2          print i
```

Μονάδες 10

2.3. Να συμπληρώσετε τα κενά στο παρακάτω πρόγραμμα, ώστε να διαβάζει δύο αριθμούς, να καλεί συνάρτηση που θα δέχεται δύο αριθμούς και θα επιστρέφει το γινόμενο τους.

```
1      def ____ (1) ____ (x,y):  
2          s= ____ (2) ____  
3          for i in range ( ____ (3) ____ )  
4              s=s+ ____ (4) ____  
5          return ____ (5) ____  
6  
7      # Κυρίως Πρόγραμμα  
8      a,b= ____ (6) ____ ('Δώσε δύο αριθμούς που θέλεις να πολλαπλασιάσω')  
9      print 'Το γινόμενο τους είναι',multi( ____ (7) ____ , ____ (8) ____ )
```

Στο πρόγραμμα υπάρχουν οκτώ (8) κενά, τα οποία έχουν αριθμηθεί και υπογραμμιστεί. Να γράψετε στο γραπτό σας τον αριθμό κάθε κενού και δίπλα τι πρέπει να συμπληρωθεί, ώστε το τμήμα προγράμματος να εκτελεί σωστά την λειτουργία του.

Μονάδες 8

ΘΕΜΑ 2

2.1

1. `ar=12.40`
2. `ar=ar+ar*40/100.`
3. `new=int(ar)`
4. `first=85/10`
5. `last=85%10`
6. `print last**2`
7. `print (first+last)/2.`

2.2

```
1      i=100
2      while i>0:
3          print i
4          i=i-2
```

2.3

```
1      multi
2      0
3      x
4      y
5      s
6      input
7      a
8      b
```

ΘΕΜΑ 4

Μια εταιρεία παρκινγκ λειτουργεί όλο το 24ωρο σε κεντρικό σημείο της Αθήνας. Η χρέωση για την στάθμευση γίνεται κλιμακωτά όπως φαίνεται στον παρακάτω πίνακα:

ΔΙΑΡΚΕΙΑ ΣΤΑΘΜΕΥΣΗΣ	ΚΟΣΤΟΣ ΑΝΑ ΩΡΑ
Μέχρι και 2 ώρες	3.5 €
3 ^η και 4 ^η ώρες	4 €
5 ^η έως 10 ^η ώρα	4.5 €
11 ^η έως 24 ^η ώρα	8 €

Να κατασκευάσετε πρόγραμμα σε Python το οποίο:

Δ1. Για κάθε αυτοκίνητο που στάθμευσε στο παρκινγκ διαβάζει τον αριθμό κυκλοφορίας μέχρι να δοθεί το XXX0000. Να θεωρήσετε ότι ο αριθμός κυκλοφορίας μπορεί να περιέχει τόσο γράμματα όσο και αριθμούς.

Μονάδες 5

Δ2. Διαβάζει τη διάρκεια στάθμευσης σε ώρες και τη δέχεται μόνο εφ' όσον είναι μεγαλύτερη από το 0.

Μονάδες 8

Δ3. Εμφανίζει τον αριθμό κυκλοφορίας και το ποσό που αναλογεί να πληρώσει ο οδηγός του αυτοκινήτου.

Μονάδες 4

Δ4. Υπολογίζει και εμφανίζει το ποσοστό των αυτοκινήτων που στάθμευσαν για πάνω από 5 ώρες.

Μονάδες 8

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```


ΘΕΜΑ 4

```
1  pla=pl5=0  #4.4 Αρχικοποίηση μεταβλητών πλήθος αυτοκινήτων , πλήθος πάνω από 5 ώρες
2  #4.1 Εισαγωγή αριθμού κυκλοφορίας
3  ar_kykl=raw_input('Δώσε τον αριθμό κυκλοφορίας του αυτοκινήτου')
4  #4.1 Έλεγχος ο αριθμός κυκλοφορίας να είναι διαφορετικός από την τιμή XXX0000
5  while ar_kykl!='XXX0000':
6      wres=float(input('Δώσε ώρες στάθμευσης'))  #4.2 Διάβασμα ωρών στάθμευσης
7  #4.2 Έλεγχος εγκυρότητας δεδομένων αν οι ώρες είναι θετικές
8      while wres<=0:
9          print 'Εδωσες λάθος ώρες'  #4.2 μήνυμα λάθους για μη αποδεκτή τιμή
10         wres=int(input('Δώσε ώρες στάθμευσης >0'))
11     pla=pla+1  #4.4 πλήθος αυτοκινήτων που στάθμευσαν
12     if wres<=2:  #4.2 Υπολογισμός κλιμακωτής χρέωσης
13         xrewsi=wres*3.5  # οι πρώτες 2 ώρες *3.5
14     elif wres<=4:
15         # οι πρώτες 2 ώρες*3.5 και οι υπόλοιπες ώρες *4
16         xrewsi=2*3.5+(wres-2)*4
17     elif wres<=10:
18         # οι πρώτες 2 ώρες*3.5 οι επόμενες 2 ώρες *4 και οι υπόλοιπες *4.5
19         xrewsi=2*3.5+2*4+(wres-4)*4.5
20     else:
21         # οι πρώτες 2 ώρες*3.5 οι επόμενες 2 ώρες *4 οι επόμενες 6 ώρες *4.5 και οι υπόλοιπες *8
22         xrewsi=2*3.5+2*4+6*4.5+(wres-10)*8
23     # 4.3
24     print 'Το αυτοκίνητο με αριθμό κυκλοφορίας ',ar_kykl,'θα πληρώσει ',xrewsi,
25     print ' € για ',wres,' ώρες στάθμευσης'
26     if wres>5:  #4.4 έλεγχος πόσα αυτοκίνητα σταθμευσαν πάνω από 5 ώρες
27         pl5=pl5+1
28     #διαβάζει επόμενο αριθμό κυκλοφορίας
29     ar_kykl=raw_input('Δώσε επόμενο αριθμό κυκλοφορίας του αυτοκινήτου ή XXX0000 για τέλος ')
30 if pla!=0 and pl5!=0:
31     pososto=pl5/float(pla)*100  #4.4 υπολογισμός ποσοστού
32     print 'Το ποσοστό των αυτοκινήτων που στάθμευσαν πάνω από 5 ώρες είναι: ',pososto,'% '
33 else:
34     print 'Δεν διαβάστηκαν δεδομένα ή δεν στάθμευσαν αυτοκίνητα πάνω από 5 ώρες'
```

ΘΕΜΑ 2

2.1. Δίνεται το παρακάτω τμήμα προγράμματος:

```
1 test=input('Παρακαλώ δώσε το βαθμό που πήρες από το τέστ')
2 if test<=20:
3     #Εμφωλευμένο if
4     if test>=18:
5         print 'Μπράβο τα πήγες πολύ καλά'
6         # για 18<=test<=20
7     else:
8         if test>=15:
9             print 'Μπράβο τα αρκετά καλά'
10            # για 15<=test<=18
11        else:
12            if test>=10:
13                print 'Τα πήγες σχετικά καλά'
14                # για 10<=test<=15
15            else:
16                print 'Θα πρέπει να διαβάσεις ξανά το κεφάλαιο'
17                # για test<10
18    else:
19        # για test>20
20    print 'Ο βαθμός που έδωσες είναι πάνω από είκοσι'
```

α). Τι πιστεύετε ότι κάνει το πρόγραμμα; Λειτουργεί σωστά για όλες τις τιμές εισόδου; Αν όχι να αιτιολογήσετε με ποια τιμή εισόδου δημιουργείτε πρόβλημα;

Μονάδες 5

β). Να τροποποιήσετε κατάλληλα το παραπάνω τμήμα προγράμματος ώστε να λειτουργεί σωστά για οποιαδήποτε τιμή εισόδου και να το ξαναγράψετε στο γραπτό σας (διορθωμένο) χρησιμοποιώντας αποκλειστικά την εντολή **if...elif...else** εξαλείφοντας το **εμφωλευμένο if**.

Μονάδες 10

2.2. Δίνεται το παρακάτω πρόγραμμα σε Python:

```
1 L=[]
2 for i in range (10):
3     L= [i] + L
4 print L
```

Να γράψετε στο γραπτό σας τι θα εμφανιστεί μετά την εκτέλεση της εντολής **print L** (αριθμός γραμμής 4).

Μονάδες 5

2.3. Στο παρακάτω τμήμα προγράμματος Python να συμπληρώστε κατάλληλα την συνθήκη της εντολής **while** (εντολή 2), ώστε να γίνεται έλεγχος εγκυρότητας τιμών εξασφαλίζοντας ότι ο χρήστης θα εισάγει τελικά μια από τις τιμές 'ναι' ή 'όχι', σε διαφορετική περίπτωση θα εμφανίζεται μήνυμα λάθους.

```
1 choice=raw_input('Σας ενδιαφέρει ο προγραμματισμός?(ναι/οχι)')
2 while _____:
3     print 'Έδωσες λάθος τιμή'
4     choice=raw_input('Παρακαλώ δώστε μια από τις τιμές ναι ή όχι')
```

Να γράψετε στο γραπτό σας την κατάλληλη συνθήκη ώστε να υλοποιεί τον αναφερόμενο έλεγχο εγκυρότητας.

Μονάδες 5

ΘΕΜΑ 2

2.1.

α). Το πρόγραμμα αυτό διαβάζει τον βαθμό ενός τεστ και εμφανίζει τον χαρακτηρισμό της επίδοσης του. Το πρόγραμμα έχει ένα λογικό λάθος. Σε περίπτωση που δοθεί για βαθμός μια αρνητική τιμή το πρόγραμμα εκτελείται κανονικά και εμφανίζει το εσφαλμένο μήνυμα «Θα πρέπει να διαβάσεις ξανά το κεφάλαιο. Μην διστάσεις να μου πεις τις απορίες σου» ενώ θα έπρεπε να βγάλει ένα μήνυμα λάθους ότι η τιμή είναι αρνητική ή μη αποδεκτή.

β). Το τμήμα προγράμματος που θα λειτουργεί σωστά για οποιαδήποτε τιμή εισόδου, χωρίς **εμφωλευμένο if** και με αποκλειστική χρήση της εντολής **if...elif...else** (πολλαπλή επιλογή) θα είναι το ακόλουθο:

```
1 test=input('Παρακαλώ δώσε το βαθμό που πήρες από το τέστ')
2 if test<0:
3     print 'Έδωσες Λάθος βαθμό '
4 elif test>20:
5     print 'Έδωσες Λάθος βαθμό '
6 elif test>=18 and test<=20:
7     print 'Μπράβο τα πήγες πολύ καλά'
8     # για 18<=test<=20
9 elif test>=15 and test<18:
10    print 'Μπράβο αρκετά καλά'
11    # για 15<=test<18
12 elif test>=10 and test<15:
13    print 'Τα πήγες σχετικά καλά'
14    # για 10<=test<15
15 else:
16    print 'Θα πρέπει να διαβάσεις ξανά το κεφάλαιο'
17    # για test<10
```

2.2.

[9, 8, 7, 6, 5, 4, 3, 2, 1, 0]

2.3.

while choice!='ναι' **and** choice!='οχι' :

ΘΕΜΑ 2

2.1 Να γράψετε στο γραπτό σας ισοδύναμο με το παρακάτω τμήμα προγράμματος Python, στο οποίο η σύνθετη if θα έχει αντικατασταθεί από δύο απλές if:

```
1  if HUMIDITY >= 0.30 and HUMIDITY <= 0.50:
2      print "Φυσιολογική υγρασία."
3  else:
4      print "Υγρασία εκτός φυσιολογικών ορίων."
```

Μονάδες 10

2.2 Το ακόλουθο πρόγραμμα σε Python έχει γραφεί για να διαβάζει από το πληκτρολόγιο διαδοχικά θετικούς ακεραίους μέχρι να διαβαστεί το 0 και να εμφανίζει στην οθόνη: α) τον μεγαλύτερο από τους αριθμούς που εισήχθησαν και β) το άθροισμα όσων ήταν πολλαπλάσια του 10. Στο πρόγραμμα, όμως, υπάρχουν πέντε (5) λάθη. Για κάθε ένα από τα λάθη να γράψετε στο γραπτό σας τον αριθμό της γραμμής στην οποία το εντοπίσατε, να περιγράψετε το λάθος και να δηλώσετε το είδος του (λογικό ή συντακτικό):

```
1  MAX = 0
2  SUM_10 = 0
3  NUM= int(input('ΔΩΣΕ ΘΕΤΙΚΟ ΑΚΕΡΑΙΟ:'))
4  while NUM != 0:
5      if NUM > MAX:
6          NUM = MAX
7      if NUM / 10 == 0:
8          SUM_10 = SUM_10 + 1
9      NUM= input(int('ΔΩΣΕ ΘΕΤΙΚΟ ΑΚΕΡΑΙΟ Ή 0 ΓΙΑ ΤΕΡΜΑΤΙΣΜΟ :'))
10 print 'Ο ΜΕΓΑΛΥΤΕΡΟΣ ΑΠΟ ΤΟΥΣ ΔΟΘΕΝΤΕΣ ΑΡΙΘΜΟΥΣ: ' , MAX
11 print 'ΑΘΡΟΙΣΜΑ ΠΟΛΛΑΠΛΑΣΙΩΝ ΤΟΥ 10: ' SUM_10
```

Μονάδες 15

ΘΕΜΑ 2

2.1 Ισοδύναμο τμήμα προγράμματος :

```
1  if HUMIDITY >= 0.30 and HUMIDITY <= 0.50:
2      print "Φυσιολογική υγρασία."
3  If HUMIDITY < 0.30 or HUMIDITY > 0.50 :
4      print "Υγρασία εκτός φυσιολογικών ορίων."
```

Εναλλακτικά:

```
1  if HUMIDITY >= 0.30 and HUMIDITY <= 0.50:
2      print "Φυσιολογική υγρασία."
3  if not(HUMIDITY >= 0.30 and HUMIDITY <= 0.50) :
4      print "Υγρασία εκτός φυσιολογικών ορίων."
```

2.2

Αριθμός γραμμής	Περιγραφή λάθους	Είδος λάθους
4	NUM = MAX Αν βρεθεί αριθμός μεγαλύτερος από τον MAX, ενημερώνεται ο MAX κι όχι ο αριθμός . <u>Σωστό</u> : MAX= NUM	Λογικό
7	if NUM / 10 == 0 . Ο τελεστής υπολοίπου της διαίρεσης γράφεται % κι όχι / <u>Σωστό</u> : if NUM % 10 == 0	Λογικό
8	SUM_10 = SUM_10 + 1. <u>Σωστό</u> : SUM_10 = SUM_10 + NUM	Λογικό
9	NUM= input(int('...')) <u>Σωστό</u> : NUM= int(input('...'))	Συντακτικό
11	Λείπει το κόμμα (,) μεταξύ μηνύματος και μεταβλητής	Συντακτικό

ΘΕΜΑ 4

Ένας Οίκος Δημοπρασιών διοργάνωσε δημοπρασία με σκοπό την πώληση 12 πινάκων ζωγραφικής γνωστού Έλληνα ζωγράφου. Οι μισοί από τους πίνακες αυτούς έχουν ως γενικό θέμα «ΤΟΠΙΟ» και οι άλλοι μισοί έχουν ως γενικό θέμα «ΠΟΡΤΡΕΤΟ». Βάσει των κανονισμών του, για κάθε αντικείμενο που πωλείται σαν αποτέλεσμα της δημοπρασίας ο Οίκος Δημοπρασιών παρακρατεί το 20% της τιμής πώλησης. Να αναπτύξετε πρόγραμμα σε γλώσσα προγραμματισμού Python το οποίο:

4.1 Για κάθε πίνακα ζωγραφικής που δημοπρατήθηκε να διαβάζει τον τίτλο του, το θέμα του (‘Τ’ για ΤΟΠΙΟ ή ‘Π’ για ΠΟΡΤΡΕΤΟ), την τιμή εκκίνησής του στη δημοπρασία σε ευρώ και την τιμή πώλησής του σε ευρώ (0 ευρώ αν δεν πουλήθηκε). Δεν χρειάζεται έλεγχος των εισαγόμενων δεδομένων ως προς την εγκυρότητά τους. Θεωρείστε πως πουλήθηκε τουλάχιστον ένας πίνακας ζωγραφικής.

Μονάδες 7

4.2 Να εντοπίζει και να εμφανίζει στην οθόνη τον τίτλο του πίνακα ζωγραφικής που είχε τη μεγαλύτερη διαφορά μεταξύ τιμής πώλησης και τιμής εκκίνησης. Θεωρείστε πως ο πίνακας αυτός είναι μοναδικός.

Μονάδες 8

4.3 Να υπολογίζει και να εμφανίζει στην οθόνη με κατάλληλο μήνυμα το πλήθος των πινάκων που πωλήθηκαν, καθώς και το συνολικό ποσό της προμήθειας που θα εισπράξει ο Οίκος Δημοπρασιών για την πώλησή τους.

Μονάδες 10

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1  MAX_DIAF= -1 # 4.2 Μέγιστη διαφορά μεταξύ τιμής πώλησης-εκκίνησης
2  sold = 0 # 4.3 Αρχικοποίηση πλήθους πινάκων που πωλήθηκαν
3  sum_sold = 0 # 4.3 Αρχικοποίηση συνολικού ποσού πωλήσεων
4
5  # 4.1: Εισαγωγή των δεδομένων
6  for i in range(12):
7      TITLE= raw_input('Δώσε τον τίτλο του πίνακα:')
8      THEMA= raw_input('Δώσε το θέμα του (Π για πορτρέτο ή Τ για τοπίο:')
9      TIMH_EKKIN = float(input('Δώσε την τιμή εκκίνησης (σε ευρώ):'))
10     TIMH_TEL = float(input('Δώσε την τιμή πώλησης (0 αν δεν πωλήθηκε):'))
11     # 4.2: Διαφορά μεταξύ τιμή πώλησης και εκκίνησης
12     DIAF = TIMH_TEL - TIMH_EKKIN
13     if DIAF > MAX_DIAF:
14         MAX_DIAF = DIAF
15         MAX_TITLE = TITLE
16     # 4.3: Αν ο πίνακας πωλήθηκε
17     if TIMH_TEL != 0:
18         sold = sold + 1
19         sum_sold = sum_sold + TIMH_TEL
20
21 # 4.2
22 print 'Πίνακας με τη μέγιστη διαφορά τιμής πώλησης-εκκίνησης:', MAX_TITLE
23 # 4.3
24 print 'Πλήθος πινάκων που πωλήθηκαν: ', sold
25 PROM= sum_sold * 20 / 100.
26 print 'Συνολικό ποσό προμήθειας για τον Οίκο Δημοπρασιών ', PROM, ' ευρώ.'
```


ΘΕΜΑ 2

2.1 Να γραφούν ως εντολές της Python οι ακόλουθες εκφράσεις:

- A.** Η τιμή της μεταβλητής Z είναι ανάμεσα στο 1 και στο 100 (χωρίς να συμπεριλαμβάνονται οι δύο ακραίες τιμές)
- B.** Η τιμή της ακέραιας μεταβλητής Y είναι άρτιος αριθμός αλλά όχι πολλαπλάσιο του 4
- Γ.** Η μεταβλητή X έχει την τιμή 1 ή την τιμή 2
- Δ.** Το τελευταίο ψηφίο της ακέραιας μεταβλητής AP είναι 5
- Ε.** Η απόλυτη τιμή της μεταβλητής K είναι 35

Μονάδες 10

2.2 Το ακόλουθο πρόγραμμα σε Python έχει γραφεί για να διαβάζει επαναληπτικά το ύψος άγνωστου πλήθους αθλητών και να υπολογίζει και να εμφανίζει στην οθόνη το μέσο ύψος τους. Η εισαγωγή των δεδομένων ολοκληρώνεται όταν καταχωρηθούν τα ύψη 50 αθλητών ή όταν δοθεί ως ύψος το 0. Στο πρόγραμμα, όμως, υπάρχουν πέντε (5) λάθη. Για κάθε ένα από τα λάθη αυτά να γράψετε στο γραπτό σας τον αριθμό της γραμμής στην οποία το εντοπίσατε, να περιγράψετε το λάθος και να δηλώσετε το είδος του λάθους (λογικό ή συντακτικό):

```
1  COUNT = 0
2  SUM = 0
3  HEIGHT= float(input('ΔΩΣΕ ΥΨΟΣ ΑΘΛΗΤΗ (σε εκ.) Ή 0 ΓΙΑ ΤΕΡΜΑΤΙΣΜΟ :'))
4  while HEIGHT <>0 or COUNT < 50:
5      SUM = SUM + HEIGHT
6      COUNT = COUNT + 1
7
8  if COUNT != 0:
9      MO = COUNT / SUM
10     print 'ΜΕΣΟ ΥΨΟΣ ΜΑΘΗΤΩΝ:' , MO
11     elif:
12         print 'ΔΕΝ ΔΟΘΗΚΑΝ ΣΤΟΙΧΕΙΑ'
```

Μονάδες 15

ΘΕΜΑ 2

2.1

A. $Z > 1$ and $Z < 100$

B. $Y \% 2 == 0$ and $Y \% 4 != 0$

Γ. $X == 1$ or $X == 2$

Δ. $AP \% 10 == 5$

Ε. $\text{abs}(K) == 35$

2.2

Αριθμός γραμμής	Περιγραφή λάθους	Είδος λάθους
4	Για να συνεχιστεί η επανάληψη πρέπει να μην έχει δοθεί το 0 και να μην έχουν καταχωρηθεί τα ύψη 50 αθλητών. Λανθασμένα, λοιπόν, χρησιμοποιείται ο λογικός τελεστής or για να συνδέσει τις δύο επιμέρους συνθήκες. Σωστό: HEIGHT <>0 and COUNT < 50	Λογικό
7	Πριν ολοκληρωθεί ο βρόχος while, πρέπει να διαβαστεί το επόμενο ύψος. Λείπει η εντολή: HEIGHT= float(input('ΔΩΣΕ ΥΨΟΣ ΑΘΛΗΤΗ (σε εκ.) Ή Ο ΓΙΑ ΤΕΡΜΑΤΙΣΜΟ :'))	Λογικό
8	Ο τελεστής του διάφορου γράφεται != κι όχι != . <u>Σωστό</u> : if COUNT != 0:	Συντακτικό
9	MO = COUNT / SUM Το μέσο ύψος υπολογίζεται διαιρώντας το συνολικό ύψος δια του πλήθους των αθλητών κι όχι το αντίθετο. <u>Σωστό</u> : MO = SUM /COUNT	Λογικό
11	elif : <u>Σωστό</u> : else :	Συντακτικό

ΘΕΜΑ 4

Ένας τραπεζικός οργανισμός διοργανώνει διαγωνισμό για την πρόσληψη 100 πτυχιούχων. Στην α' φάση του διαγωνισμού συμμετέχουν 350 υποψήφιοι και η βαθμολογία τους υπολογίζεται αθροιστικά βάσει των κριτηρίων του ακόλουθου πίνακα. Στην επόμενη φάση του διαγωνισμού θα προκριθούν όσοι συγκέντρωσαν κατά την α' φάση περισσότερους από 10 βαθμούς.

Ανώτατος τίτλος σπουδών		Εργασιακή εμπειρία στα χρηματοοικονομικά	
Είδος πτυχίου	Βαθμός	Έτη προϋπηρεσίας	Βαθμός
΄Δ΄ για Διδακτορικό Δίπλωμα	5	Τουλάχιστον 10	2 βαθμοί ανά έτος προϋπηρεσίας
΄Μ΄ για Μεταπτυχιακό Δίπλωμα	3	3-9	1 βαθμός ανά έτος προϋπηρεσίας
΄Π΄ για Βασικό Πτυχίο	1	0-2	0

Να αναπτυχθεί πρόγραμμα σε Python το οποίο:

4.1 Να διαβάσει για κάθε υποψήφιο τον κωδικό υποψηφίου (1-350), τον ανώτατο τίτλο σπουδών του (΄Δ΄, ΄Μ΄, ή ΄Π΄) και τα έτη προϋπηρεσίας του στον χρηματοοικονομικό τομέα. Δεν χρειάζεται έλεγχος των εισαγόμενων δεδομένων ως προς την εγκυρότητά τους.

Μονάδες 5

4.2 Να υπολογίζει τη συνολική βαθμολογία κάθε υποψηφίου, με βάση τα δοσμένα κριτήρια. Να ελέγχει αν ο υποψήφιος περνά στη β' φάση του διαγωνισμού ή όχι και να εμφανίζει ανάλογο μήνυμα.

Μονάδες 14

4.3 Να υπολογίζει και να εμφανίζει με κατάλληλο μήνυμα το πλήθος των υποψηφίων που δεν πέρασαν στην επόμενη φάση του διαγωνισμού.

Μονάδες 6

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1  pl = 0 # 4.3 Αρχικοποίηση πλήθους υποψηφίων που δεν προκρίνονται
2
3  for i in range(350): # 4.1: εισαγωγή των δεδομένων
4      KOD = int(input('Δώσε τον κωδικό του υποψηφίου (1-350):'))
5      DEGREE = raw_input('Δώσε τον ανώτατο τίτλο σπουδών του (Δ, Μ ή Π):')
6      EXPER = int(input('Δώσε τα έτη προϋπηρεσίας του στα χρηματοοικονομικά:'))
7
8  # 4.2: υπολογισμός συνολικής βαθμολογίας υποψηφίου
9      if DEGREE == 'Δ':
10         B = 5
11     elif DEGREE == 'Μ':
12         B = 3
13     else: # έχει το Βασικό Πτυχίο
14         B = 1
15
16     if EXPER >= 10:
17         B = B + EXPER * 2
18     elif EXPER >= 3:
19         B = B + EXPER
20
21     if B > 10: # 4.2
22         print 'Ο υποψήφιος με κωδικό ', KOD, ' περνά στη β φάση του διαγωνισμού'
23     else:
24         print 'Ο υποψήφιος με κωδικό ', KOD, ' δεν περνά στη β φάση .'
25         pl = pl + 1 # 4.3 αύξηση υποψηφίων που δεν προκρίνονται
26
27 print 'Πλήθος υποψηφίων που δεν πέρασαν στην επόμενη φάση: ', pl # 4.3
```

ΘΕΜΑ 2

2.1. Δίνεται το παρακάτω τμήμα προγράμματος στη γλώσσα προγραμματισμού Python. Να γράψετε στο γραπτό σας τα αποτελέσματα που θα εμφανιστούν αν δοθεί σαν είσοδος η τιμή 7.

```
1      x=int(input('Δώσε ακέραιο αριθμό: '))
2      print (x % 3)
3      x=x / 3
4      print x
5      x=x+(x % 3)
6      print x
7      x=x /3
8      print x
```

Μονάδες 12

2.2. Δίνεται το παρακάτω τμήμα προγράμματος στη γλώσσα προγραμματισμού Python. Να μετατραπεί σε ισοδύναμο τμήμα προγράμματος, σε γλώσσα προγραμματισμού Python, χρησιμοποιώντας **αποκλειστικά** την δομή επανάληψης while:

```
1      s=0
2      for i in range (1,10, 2):
3          s=s+i
4      print s
```

Μονάδες 13

ΘΕΜΑ 2

2.1.

Γραμμή 2: 1

Γραμμή 4: 2

Γραμμή 6: 4

Γραμμή 8: 1

2.2.

Το τμήμα προγράμματος με την μετατροπή θα είναι:

```
1    s=0
2    i=1
3    while i<10:
4        s=s+i
5        i=i+2
6    print s
```

ΘΕΜΑ 4

Το ωράριο ενός εργαζομένου στην ναυπηγική ζώνη της Samsung στην Ιαπωνία είναι 38 ώρες την εβδομάδα και το ωρομίσθιο είναι 9.80€. Κάθε εργαζόμενος έχει την δυνατότητα να εργαστεί υπερωριακά μέχρι και 30 ώρες την εβδομάδα, σύμφωνα με τις εργασιακό πλαίσιο της χώρας. Για τις **υπερωρίες** κάθε εργαζόμενος αμείβεται **κλιμακωτά** σύμφωνα με τον παρακάτω πίνακα:

Ώρες υπερωρίας	Ωρομίσθιο υπερωρίας
1-19	12€
20-25	15€
26-30	18€

Οι παραπάνω αμοιβές υπόκεινται σε φορολογία 15%. Να αναπτύξετε πρόγραμμα στην γλώσσα προγραμματισμού Python το οποίο

4.1. Να διαβάζει για **έναν** εργαζόμενο το όνομα και τις συνολικές ώρες εργασίας του κατά την διάρκεια μιας εβδομάδας, **ελέγχοντας** αν οι ώρες εργασίας είναι αριθμός μεγαλύτερος ή ίσος του 38 και μικρότερος ή ίσος του 68. Σε περίπτωση λανθασμένης εισαγωγής να ζητά ξανά τον αριθμό ωρών ανά εβδομάδα.

Μονάδες 6

4.2. Να υπολογίζει την μεικτή εβδομαδιαία αποζημίωση του εργαζομένου, λαμβάνοντας υπόψη το ωρομίσθιο του κανονικού ωραρίου και το ωρομίσθιο των υπερωριών.

Μονάδες 10

4.3. Να υπολογίζει τον φόρο που αντιστοιχεί στην συνολική εβδομαδιαία αποζημίωση του εργαζομένου και τις καθαρές αποδοχές του.

Μονάδες 5

4.4. Να εμφανίζει με κατάλληλο μήνυμα το όνομα του εργαζομένου και τις μεικτές εβδομαδιαίες αποδοχές του. Επίσης σε άλλη γραμμή τον φόρο που αντιστοιχεί στις μεικτές εβδομαδιαίες αποδοχές και τις καθαρές αποδοχές του.

Μονάδες 4

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
    print b
```


ΘΕΜΑ 4

```
1  mixd=0 # 4.2 αρχικοποίηση μεικτής αποζημίωσης
2  #4.1 Διαβάζει το όνομα του εργαζόμενου
3  name=raw_input('Δώσε το όνομα του εργαζόμενου: ')
4  #4.1 Διαβάζει τις ώρες εργασίας μιας εβδομάδας
5  wres=int(input('Δώσε τις ώρες εργασίας μιας εβδομάδας: '))
6  #4.1 Ελέγχει αν οι ώρες εργασίας πληρούν τις προϋποθέσεις του προβλήματος
7  while wres<38 or wres>68:
8      wres=int(input('Δώσε σωστά τις ώρες εργασίας μιας εβδομάδας (38-68): '))
9  #4.2 Υπολογίζει την μεικτή εβδομαδιαία αποζημίωση
10  misthos=38*9.80 #4.2 Υπολογίζει τον βασικό μισθό
11  wres_yper=wres-38 #4.2 Υπολογίζει τις ώρες υπερωρίας
12  #4.2 Υπολογίζει τον μισθό των υπερωριών κλιμακωτά
13  if wres_yper !=0:
14      if wres_yper<=19:
15          misthos_yper=wres_yper*12
16      elif wres_yper<=25:
17          misthos_yper=19*12+ (wres_yper-19)*15
18      else:
19          misthos_yper=19*12+6*15+ (wres_yper-25)*18
20  mixd=misthos + misthos_yper
21  #4.3 Υπολογισμός φόρου
22  foros=mixd*15/100
23  #4.3 Υπολογισμός καθαρού μισθού
24  teliko_poso=mixd-foros
25  #4.4 Εμφανίσεις ονόματος, μεικτού μισθού, φόρου και καθαρού μισθού
26  print 'Ο εργαζόμενος: ', name, ' έχει μεικτό μισθό: ', mixd, 'ευρώ'
27  print 'Του αναλογεί φόρος: ',foros, 'ευρώ και έχει καθαρές αποδοχές: ', teliko_poso,'ευρώ'
```

ΘΕΜΑ 2

2.1. Δίνεται το παρακάτω τμήμα προγράμματος στη γλώσσα προγραμματισμού Python. Τι τιμή θα εμφανιστεί κατά την εκτέλεσή του;

```
1    sum = 0
2    for i in range(1, 8,3):
3        sum = sum + i **2
4    print 'Το άθροισμα είναι: ',sum
```

Μονάδες 6

2.2. Στον παρακάτω πίνακα η στήλη A περιέχει αριθμητικές πράξεις (παραστάσεις) ενώ η στήλη B περιέχει την αποτίμησή τους (αποτελέσματα τους) στο IDLE της Python. Να γράψετε στο τετράδιό σας τους αριθμούς 1,2,3,4,5 της **Στήλης A** και δίπλα το γράμμα α, β, γ, δ, ε της **Στήλης B**, που δίνει τη σωστή αντιστοιχία.

Στήλη A	Στήλη B
1. 5%6	α. 3
2. 7%3-3**2	β. 1
3. 8/5*3	γ. 5
4. 3*(7%2)-6/(1+1)	δ. -8
5. 3**2%2**2	ε. 0
	στ. 12

Μονάδες 10

2.3. Δίνεται το παρακάτω τμήμα προγράμματος στη γλώσσα προγραμματισμού Python. Να μετατραπεί σε ισοδύναμο τμήμα προγράμματος, σε γλώσσα προγραμματισμού Python, χρησιμοποιώντας αποκλειστικά την δομή επανάληψης **for**.

```
1    X=13
2    while X<=20 :
3        print X
4        X =X+2
5    print X+2
```

Μονάδες 9

ΘΕΜΑ 2

2.1.

Θα εμφανιστεί το 66, αφού θα γίνουν 3 επαναλήψεις με τιμές του μετρητή 1, 4, 7.

Άρα $\text{sum}=1+16+49=66$

2.2.

Οι αντιστοιχίες είναι:

1-γ

2-δ

3-α

4-ε

5-β

2.3. Ενδεικτική λύση που δίνει τις εκτυπώσεις 13,15,17,19,21, όπως το αρχικό τμήμα προγράμματος

```
1   for X in range (13,21,2):  
2       print X  
3       print X+2
```

ΘΕΜΑ 4

Ένα κατάστημα αθλητικών ειδών προσφέρει στους πελάτες του τα παρακάτω ποσοστά έκπτώσεων, ανάλογα με τον τρόπο πληρωμής που θα επιλέξει ο πελάτης:

Τρόπος πληρωμής	Ποσοστό έκπτωσης (%)
Μετρητά	15
Κάρτα	7

Να αναπτύξετε πρόγραμμα στην γλώσσα προγραμματισμού Python το οποίο:

4.1. Να διαβάζει για **έναν** πελάτη την συνολική αξία των αθλητικών ειδών που αγόρασε και τον τρόπο πληρωμής. Η αξία να είναι θετικός αριθμός και ο τρόπος πληρωμής **cash** αν πληρώνει μετρητοίς ή **card** αν πληρώνει με χρήση κάρτας. Να ελέγχει αν η αξία είναι θετικός αριθμός και ο τρόπος πληρωμής cash ή card. Σε περίπτωση λανθασμένης εισαγωγής της αξίας ή του τρόπου πληρωμής, να ζητά ξανά την αξία ή τον τρόπο πληρωμής.

Μονάδες 10

4.2. Να υπολογίζει και να εμφανίζει το ποσό της έκπτωσης ανάλογα με τον τρόπο πληρωμής.

Μονάδες 6

4.3. Να υπολογίζει και να εμφανίζει, το τελικό ποσό πληρωμής.

Μονάδες 4

4.4. Στην περίπτωση που το τελικό ποσό πληρωμής είναι μεγαλύτερο των 280 ευρώ να εμφανίζει το μήνυμα *‘Κερδίσατε δώρο 10 ευρώ στην επόμενη αγορά σας’*.

Μονάδες 5

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1  #4.1 Εισαγωγή και έλεγχος αξίας αγορών
2  axia=float(input(' Δώσε την συνολική αξία των αγορών (>0): '))
3  while axia<=0:
4      axia=float(input(' Δώσε σωστά την συνολική αξία των αγορών (>0): '))
5  #4.1 Εισαγωγή και έλεγχος τρόπου πληρωμής
6  tropos=raw_input(' Δώσε τρόπο πληρωμής (cash/card): ')
7  while tropos !='cash' and tropos!='card':
8      tropos=raw_input(' Δώσε σωστά τον τρόπο πληρωμής (cash/card): ')
9  #4.2 Υπολογισμός και εμφάνιση ποσού έκπτωσης
10 if tropos =='cash':
11     ekpt=axia*15/100
12 else:
13     ekpt=axia*7/100
14 print 'Η έκπτωση είναι: ', ekpt
15 #4.3 Υπολογισμός και εμφάνιση τελικού ποσού πληρωμής
16 teliko=axia-ekpt
17 print 'Η τελική αξία των προϊόντων είναι: ', teliko
18 #4.4 Έλεγχος τελικού ποσού και εμφάνιση μηνύματος
19 if teliko > 280:
20     print 'Κερδίσατε δώρο 10 ευρώ στην επόμενη αγορά σας'
21 else:
22     print ' Ευχαριστούμε για τις αγορές σας'
```

ΘΕΜΑ 2

2.1. Δίνονται οι τιμές των μεταβλητών A=2, B=1, C=3, D=4 και η παρακάτω σύνθετη λογική έκφραση:

not (C*4 % 2 == B(C-2) and (D== 8 / A)) or (A +B**0 ==3)**

Να αντικαταστήσετε τις μεταβλητές με τις τιμές που δίνονται, να εκτελέσετε τις αριθμητικές πράξεις, να κάνετε τις απαραίτητες συγκρίσεις και να εκτελέσετε τις λογικές πράξεις, ώστε να υπολογίσετε την τελική τιμή της έκφρασης.

Μονάδες 5

2.2. Να σχεδιάσετε τον πίνακα τιμών και τις τιμές που θα εμφανιστούν στην οθόνη μετά την εκτέλεση του παρακάτω τμήματος προγράμματος σε γλώσσα προγραμματισμού Python

```
1   x =2
2   for k in range(15,1,-2):
3       x= x + k
4       print x
5       print x%2
```

Μονάδες 14

2.3 Συμπληρώστε την συνθήκη επιλογής στο παρακάτω πρόγραμμα σε γλώσσα Python έτσι ώστε να εμφανίζει αν ένας μαθητής περνάει την τάξη ή όχι σύμφωνα με τα αποτελέσματα των γραπτών του σε 5 μαθήματα. Ο μαθητής περνάει την τάξη αν ο μέσος όρος των βαθμών του στα 5 μαθήματα είναι τουλάχιστον 10 αλλά δεν πρέπει σε κανένα μάθημα να έχει βαθμό κάτω από 8.

```
1   a = int(input('Δώσε βαθμό για το πρώτο μάθημα: '))
2   b = int(input('Δώσε βαθμό για το δεύτερο μάθημα: '))
3   c = int(input('Δώσε βαθμό για το τρίτο μάθημα: '))
4   d =int(input('Δώσε βαθμό για το τέταρτομάθημα: '))
5   e = int(input('Δώσε βαθμό για το πέμπτο μάθημα: '))
6   mo = (a+b+c+d+e)/5
7   if (_____):
8       print ' Πέρασες την τάξη'
9   else:
10      print 'Δεν πέρασες την τάξη'
```

Μονάδες 6

ΘΕΜΑ 2

2.1. Η τιμή της παράστασης είναι

True

2.2.

Οι τιμές είναι:

17

1

30

0

41

1

50

0

57

1

62

0

65

1

2.3. Η συνθήκη θα είναι όπως παρακάτω:

if (mo >= 10 and a>=8 and b>=8 and c>=8 and d>=8 and e>=8)

ΘΕΜΑ 4

Το γραπτό ενός μαθητή στις τελικές εξετάσεις αξιολογείται αρχικά από δυο βαθμολογητές και ο τελικός βαθμός είναι ο μέσος όρος τους. Αν η διαφορά των δύο βαθμολογητών είναι από 2 και πάνω μονάδες αξιολογείται και από τρίτο βαθμολογητή. Στην περίπτωση αυτή ο τελικός βαθμός είναι ο μέσος όρος των τριών βαθμολογητών.

Να αναπτύξετε πρόγραμμα υπολογισμού του τελικού βαθμού στην γλώσσα προγραμματισμού Python το οποίο:

4.1. Να διαβάζει τους βαθμούς του πρώτου και του δεύτερου. Οι βαθμοί πρέπει να είναι μεταξύ 1 και 20. Σε περίπτωση που η εισαγωγή κάποιου βαθμού είναι έξω από τα επιτρεπτά όρια να εισάγεται εκ νέου σωστός βαθμός.

Μονάδες 7

4.2. Να υπολογίζει αν το γραπτό χρειάζεται αναβαθμολόγηση. Αν το γραπτό χρειάζεται αναβαθμολόγηση θα διαβάζει και τον βαθμό του τρίτου βαθμολογητή, που επίσης πρέπει να είναι μεταξύ 1 και 20. Σε περίπτωση που ο βαθμός είναι έξω από τα επιτρεπτά όρια να εισάγεται εκ νέου σωστός βαθμός.

Μονάδες 10

4.3. Να υπολογίζει και να εμφανίζει το τελικό βαθμό του γραπτού σύμφωνα με την παραπάνω διαδικασία.

Μονάδες 8

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές.

Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```


ΘΕΜΑ 4

```
1  #4.1 Εισαγωγή και έλεγχος των βαθμών των 2 πρώτων βαθμολογητών
2  grade1=int(input('Δώσε βαθμό 1ου βαθμολογητή (1-20): '))
3  while grade1<1 or grade1>20:
4      grade1=int(input('Δώσε σωστά το βαθμό 1ου βαθμολογητή (1-20): '))
5  grade2=int(input('Δώσε βαθμό 2ου βαθμολογητή (1-20): '))
6  while grade2<1 or grade2>20:
7      grade2=int(input('Δώσε σωστά το βαθμό 2ου βαθμολογητή (1-20): '))
8
9  #4.2 Υπολογισμός για την ανάγκη αναβαθμολόγησης
10 if abs(grade1-grade2)>2:
11     #4.2 Εισαγωγή βαθμού 3ου βαθμολογητή στην περίπτωση Yes
12     grade3=int(input('Δώσε βαθμό 3ου βαθμολογητή (1-20): '))
13     while grade3<1 or grade3>20:
14         grade3=int(input('Δώσε σωστά το βαθμό 3ου βαθμολογητή (1-20): '))
15
16 #4.3 Υπολογισμός του τελικού βαθμού
17 if abs(grade1-grade2)>2:
18     telikos=(grade1+grade2+grade3)/3.
19 else:
20     telikos=(grade1+grade2)/2.
21 #4.4 Εμφάνιση τελικού βαθμού
22 print ' Ο τελικός βαθμός είναι: ', telikos
```

ΘΕΜΑ 2

2.1. Δίνεται το παρακάτω τμήμα προγράμματος στη γλώσσα προγραμματισμού Python. Τι τιμές θα εμφανιστούν κατά την εκτέλεση του;

```
1   a=0
2   while a <= 12:
3       for i in range(1,4):
4           a=a + i
5           print a
6       a=a + 5
7       print a
```

Μονάδες 13

2.2. Δίνεται η παρακάτω λίστα στην γλώσσα προγραμματισμού Python **A=[0,1,1,2,2,3]**. Γράψτε στο τετράδιό σας ποια θα είναι η μορφή (περιεχόμενα) της λίστας A ή τι θα συμβεί, μετά την εφαρμογή **κάθε** μίας από τις παρακάτω εντολές:

```
Εντολή 1: >>>A[0]=1
Εντολή 2: >>>A[1]=0
Εντολή 3: >>>A[5]=5
Εντολή 4: >>>A[4]=3
Εντολή 5: >>>A[3]=4
Εντολή 6: >>>A[6]=6
```

Μονάδες 12

ΘΕΜΑ 2

2.1. Οι τιμές που θα εμφανιστούν είναι:

1
3
6
11
12
14
17
22

2.2.

Εντολή 1: [1,1,1,2,2,3]

Εντολή 2: [1,0,1,2,2,3]

Εντολή 3: [1,0,1,2,2,5]

Εντολή 4: [1,0,1,2,3,5]

Εντολή 5: [1,0,1,4,3,5]

Εντολή 6: Μήνυμα λάθους, αφού το A[6] είναι έξω από τα όρια της λίστας.

2.3. Το τμήμα προγράμματος θα είναι:

```
1     a=int(input('Δώσε αριθμό a: '))
2     if a >= 0 and a<= 20:
3         print a
4
5     if a<0 or a>20:
6         print 'Λάθος'
```

ΘΕΜΑ 4

Οι μαθητές/τριες που επιθυμούν να επιλεγούν για τις Ένοπλες Δυνάμεις εξετάζονται σε μια σειρά αθλημάτων. Μεταξύ άλλων εξετάζονται στα παρακάτω αθλήματα, όπου, για να είναι επιτυχημένη η δοκιμασία τους (ανεξαρτήτως φύλου), πρέπει να ικανοποιούνται τα εξής κριτήρια:

- i). δρόμος 100 μέτρων, έως 16 δευτερόλεπτα,
- ii). άλμα εις ύψος με φόρα, τουλάχιστον 1 μέτρο και 5 εκατοστά.

Να αναπτύξετε πρόγραμμα στην γλώσσα προγραμματισμού Python το οποίο:

4.1. Να διαβάζει για **κάθε** μαθητή/τρια το όνομα, το φύλο και τις επιδόσεις του/της στον δρόμο 100 μέτρων σε δευτερόλεπτα και στο άλμα εις ύψος με φόρα σε εκατοστά. Η επανάληψη να σταματά όταν δοθεί σαν όνομα μαθητή/τριας ο κενός χαρακτήρας. Δεν απαιτείται έλεγχος έγκυρων τιμών.

Μονάδες 5

4.2. Να υπολογίζει και να εμφανίζει το ποσοστό των μαθητών και των μαθητριών αντίστοιχα που πέτυχαν **και** στις 2 αθλητικές δοκιμασίες.

Μονάδες 8

4.3. Να υπολογίζει και να εμφανίζει με κατάλληλο μήνυμα το όνομα και την επίδοση του μαθητή που έκανε τον καλύτερο χρόνο στον δρόμο των 100 μέτρων και πέτυχε σε όλες τις αθλητικές δοκιμασίες. Υποθέστε ότι υπάρχει μόνο ένας μαθητής με τον καλύτερο χρόνο, που πέτυχε σε όλες τις αθλητικές δοκιμασίες.

Μονάδες 6

4.4. Να υπολογίζει και να εμφανίζει με κατάλληλο μήνυμα το όνομα και την επίδοση της μαθήτριας που έκανε την καλύτερη επίδοση στο άλμα εις μήκος με φόρα και πέτυχε σε όλες τις αθλητικές δοκιμασίες. Υποθέστε ότι υπάρχει μόνο μια μαθήτρια με την καλύτερη επίδοση στο άλμα εις μήκος με φόρα, που πέτυχε σε όλες τις αθλητικές δοκιμασίες.

Μονάδες 6

Να θεωρήσετε ότι υπάρχει τουλάχιστον ένας μαθητής και μια τουλάχιστον μαθήτρια που έχουν πετύχει στις αθλητικές δοκιμασίες.

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές.

Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
    print b
```

ΘΕΜΑ 4

```
1  # Αρχικοποίηση μεταβλητών για το πρόγραμμα
2  man=0 #Όλοι οι μαθητές
3  woman=0 # Όλες οι μαθήτριες
4  man_ok=0 # Μαθητές που πέτυχαν
5  woman_ok=0 # Μαθήτριες που πέτυχαν
6  xr100=10000 #4.3 αρχικοποίηση χρόνου
7  name_xr100='' #4.3 αρχικοποίηση ονόματος με μικρότερο χρόνο
8  m_mikos=0 # 4.4 αρχικοποίηση μήκους
9  name_mikos='' #4.4 αρχικοποίηση ονόματος με την μέγιστη επίδοση
10
11 name=raw_input('Δώσε όνομα μαθητή/τριας: ')
12 while name!=' ' :
13     #4.1 Εισαγωγή δεδομένων
14     filo=raw_input('Δώσε φύλο M (αγόρι) / W (κορίτσι): ')
15     ep100=int(input('Δώσε επίδοση στα 100 μ. σε sec: '))
16     ep_mikos=int(input('Δώσε επίδοση στο ύψος σε cm: '))
17     #4.2 Υπολογισμός συμμετοχών για μαθητές και μαθήτριες
18     if filo=='W':
19         woman=woman+1
20     else:
21         man=man+1
22     #4.2 Υπολογισμός επιτυχόντων μαθητών και μαθητριών
23     if ep100<=16 and ep_mikos>=105:
24
25         if filo=='M':
26             man_ok=man_ok+1
27     #4.3 Εύρεση μαθητή με τον καλύτερο χρόνο (ελάχιστο)
28     if ep100 <xr100:
29         xr100=ep100
30         name_xr100=name
31     else:
32         woman_ok=woman_ok+1
33     #4.4 Εύρεση μαθήτριας με μεγαλύτερη επίδοση στο μήκος
34     if ep_mikos>m_mikos:
35         m_mikos=ep_mikos
36         name_mikos=name
37     name=raw_input('Δώσε νέο όνομα μαθητή/τριας: ')
38 # 4.2, 4.3, 4.4 Εμφανίσεις ζητούμενων
39 print 'Ποσοστό μαθητών που πέτυχαν στις δοκιμασίες: ', man_ok/man*100,'% '
40 print 'Ποσοστό μαθητριών που πέτυχαν στις δοκιμασίες: ', woman_ok/woman*100,'% '
41 print 'Ο μαθητής',name_xr100, ' έκανε τον καλύτερο χρόνο στα 100μ', xr100,'sec'
42 print 'Η μαθήτρια',name_mikos , 'είχε την καλύτερη επίδοση στο μήκος',m_mikos,'cm'
```

ΘΕΜΑ 2

2.1. Δίνεται το παρακάτω τμήμα προγράμματος στη γλώσσα προγραμματισμού Python. Τι τιμές θα εμφανιστούν κατά την εκτέλεση του;

```
1   a=0
2   while a<=22:
3       if a %2 ==0:
4           print a
5       else:
6           print a+1
7       a=a+3
```

Μονάδες 8

2.2. Δίνεται το παρακάτω τμήμα προγράμματος στην γλώσσα προγραμματισμού Python. Να μετατραπεί σε ισοδύναμο τμήμα προγράμματος Python με την χρήση **μόνο μίας απλής** δομής επιλογής if:

```
1   a=int(input('Δώσε ακέραιο αριθμό: '))
2   k=0
3   if a<=9:
4       k=a+3
5       a=a+1
6   else:
7       k=a+3
8   print a
9   print k
```

Μονάδες 8

2.3. Να συμπληρωθούν οι αριθμοί 1-9 έτσι ώστε το παρακάτω τμήμα προγράμματος σε Python, να **δημιουργεί μια συνάρτηση** η οποία δέχεται ως είσοδο έναν ακέραιο αριθμό N και επιστρέφει το άθροισμα και το πλήθος **όλων** των περιττών αριθμών από το 1 **έως και** το N:

```
1   __1__ perittoi (N):
2       sum = 0
3       count = 0
4       for i in range(1, __2__):
5           if(__3__):
6               sum = __4__ + __5__
7               count = __6__ + __7__
8       return __8__, __9__
```

Μονάδες 9

ΘΕΜΑ 2

2.1. Οι τιμές που θα εμφανιστούν είναι:

0
4
6
10
12
16
18
22

2.2.

```
1  a=int(input('Δώσε ακέραιο αριθμό: '))
2  k=0
3  if a<=9:
4      a=a+1
5      k=a+3
6  print a
7  print k
```


2.3. Το τμήμα προγράμματος θα είναι:

```
1    def perittoi (N):  
2        sum = 0  
3        count = 0  
4        for i in range(1, N+1):  
5            if(i % 2 == 1):  
6                sum = sum + i  
7                count = count + 1  
8        return sum, count
```

Δηλαδή:

```
1→def  
2→N+1  
3→ i%2==1  
4→ sum  
5→i  
6→count  
7→1  
8→sum  
9→count
```

ΘΕΜΑ 4

Στους 18 αθλητές μιας ομάδας μπάσκετ γίνεται καταγραφή του ύψους τους για στατιστικούς λόγους.

Να αναπτύξετε πρόγραμμα στην γλώσσα προγραμματισμού Python το οποίο:

4.1. Να δημιουργεί λίστα HIGH 18 θέσεων και να εισάγει σε αυτήν για **κάθε** αθλητή το ύψος του **σε εκατοστά**. Το ύψος πρέπει να είναι θετικός αριθμός και μικρότερος του 240. Σε περίπτωση που η εισαγωγή κάποιου ύψους είναι έξω από τα επιτρεπτά όρια να εισάγεται εκ νέου σωστός αριθμός.

Μονάδες 6

4.2. Να υπολογίζει και να εμφανίζει τον μέσο όρο ύψους της συγκεκριμένης ομάδας.

Μονάδες 6

4.3. Να υπολογίζει και να εμφανίζει με κατάλληλο μήνυμα το μεγαλύτερο ύψος που έχει εισαχθεί και πόσοι αθλητές έχουν αυτό το ύψος;

Μονάδες 7

4.4. Να υπολογίζει και να εμφανίζει με κατάλληλο μήνυμα πόσοι αθλητές έχουν ύψος μεγαλύτερο από τον μέσο όρο.

Μονάδες 6

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές.

Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
    print b
```

ΘΕΜΑ 4

```
1  #4.1 αρχικοποίηση λίστας και αθροιστή ύψους για μέσο όρο
2  HIGH=[]
3  sum=0
4  #4.1 Εισαγωγή υψών αθλητών με ταυτόχρονο έλεγχο ορίων
5  for i in range (18):
6      x=int(input('Δώσε ύψος παίκτη σε εκατοστά: '))
7      while x<=0 or x>240:
8          x=int(input('Δώσε σωστά ύψος παίκτη σε εκατοστά: '))
9      HIGH.append(x)
10     sum=sum+x
11 #4.2 Υπολογισμός και εμφάνιση μέσου όρου ύψους
12 mo=sum/18.
13 print ' Ο μέσος όρος ύψους της ομάδας είναι: ',mo
14 #4.4 Αρχικοποίηση μετρητή για υψη > μέσου όρου
15 plithos=0
16 max=-1 #4.3 αρχικοποίηση μεταβλητής για μεγαλύτερο ύψος
17 for i in range(18):
18     #4.4 Υπολογισμός πλήθους αθλητών με ύψος > μέσου όρου
19     if HIGH[i]>mo:
20         plithos =plithos+1
21 #Υπολογισμός μεγαλύτερου ύψους
22     if HIGH[i]>max:
23         max=HIGH[i]
24 #4.3 Υπολογισμός πλήθους αθλητών με το μέγιστο ύψος
25 pl=0
26 for i in range(18):
27     if HIGH[i]==max:
28         pl=pl+1
29 print 'Το μεγαλύτερο ύψος είναι: ', max,' και το έχουν',pl,' αθλητές'
30 print 'Ύψος μεγαλύτερο του μέσου όρου έχουν', plithos,' αθλητές'
```

ΘΕΜΑ 2

2.1. Υπάρχουν διάφοροι τρόποι αναπαράστασης ενός αλγόριθμου. Στην **Στήλη Α** του παρακάτω πίνακα καταγράφονται οι συνηθέστεροι τρόποι αναπαράστασης του αλγόριθμου και στην **Στήλη Β** οι περιγραφές τους. Να γράψετε στο τετράδιό σας τους αριθμούς της **Στήλης Α** και δίπλα το γράμμα της **Στήλης Β**, που δίνει τη σωστή αντιστοιχία. Επισημαίνεται ότι η **Στήλη Β** έχει και μια λάθος περιγραφή.

Στήλη Α	Στήλη Β
1. Φυσική Γλώσσα	α. Όταν ένα πρόγραμμα είναι γραμμένο είτε σε μία ψευδογλώσσα είτε σε κάποια γλώσσα προγραμματισμού.
2. Κωδικοποίηση	β. Αφορά μια υποθετική δομημένη γλώσσα με στοιχεία από υπαρκτές γλώσσες προγραμματισμού, με λίγες εντολές και απλοποιημένη σύνταξη.
3. Ψευδογλώσσα ή Ψευτοκώδικας	γ. Αφορά αναπαραστάσεις οι οποίες συνιστούν ένα γραφικό τρόπο παρουσίασης του αλγόριθμου
4. Διαγραμματικές τεχνικές	δ. Αποτελεί τον πιο απλό, ανεπεξέργαστο και αδόμητο τρόπο παρουσίασης ενός αλγόριθμου, που με απλά λόγια και ελεύθερες εκφράσεις, περιγράφουμε τα βήματά του
	ε) Αφορά λογικές παραστάσεις που υπολογίζονται με βάση μια υποθετικά νοηματοδοτούμενη γλώσσα, η οποία σηματοδοτεί την γραμμική της διάσταση

2.2. Δίνεται το παρακάτω τμήμα προγράμματος στη γλώσσα προγραμματισμού Python. Να μετατρέψετε την σύνθετη δομή επιλογής (if...else) σε 2 απλές (if) που να έχουν το ίδιο αποτέλεσμα.

```
1    a=int(input('Δώσε αριθμό a: '))
2    if a >= 0 and a<= 20:
3        print a
4    else:
5        print 'Λάθος'
```

Μονάδες 13

ΘΕΜΑ 2

2.1.

Οι σωστές αντιστοιχίσεις είναι: **1-δ, 2-α, 3-β, 4-γ**

2.2.

```
1      a=int(input('Δώσε αριθμό a: '))
2      if a >= 0 and a<= 20:
3          print a
4
5      if a<0 or a>20:
6          print 'Λάθος'
```

ΘΕΜΑ 4

Για την απόκτηση του κρατικού Πιστοποιητικού Γλωσσομάθειας, στο ανώτατο επίπεδο, στην εξέταση της Γαλλικής Γλώσσας εξετάζονται 250 υποψήφιοι. Η εξέταση περιλαμβάνει προφορική και γραπτή δοκιμασία με βαθμολογία και στις δύο περιπτώσεις από το 1 έως το 100 για κάθε εξέταση. Ο υποψήφιος για να είναι επιτυχών πρέπει ο μέσος όρος των δύο βαθμολογιών του να είναι μεγαλύτερος ή ίσος του 60.

Να αναπτύξετε πρόγραμμα στην γλώσσα προγραμματισμού Python το οποίο:

4.1. Να διαβάζει το όνομα, την προφορική και τη γραπτή βαθμολογία **κάθε** υποψηφίου. Η κάθε βαθμολογία πρέπει να είναι μεγαλύτερη του 0 και μικρότερη ή ίση του 100. Σε περίπτωση που η εισαγωγή κάποιας βαθμολογίας είναι έξω από τα επιτρεπτά όρια να εισάγεται εκ νέου σωστή βαθμολογία.

Μονάδες 8

4.2. Να ελέγχει τις δύο βαθμολογίες κάθε υποψηφίου και να εμφανίζει το όνομά του και κατάλληλο μήνυμα σε σχέση με το αν η προφορική βαθμολογία είναι μεγαλύτερη, μικρότερη ή ίση σε σχέση με την γραπτή.

Μονάδες 8

4.3. Να υπολογίζει και να εμφανίζει με κατάλληλο μήνυμα το ποσοστό των υποψηφίων που πέρασαν με επιτυχία τις εξετάσεις.

Μονάδες 9

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές.

Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1  #Αρχικοποίηση μετρητή επιτυχόντων
2  petixe=0
3  #4.1 Επαναληπτικά εισαγωγή και έλεγχος προφορικής και γραπτής βαθμολογίας
4  for i in range (250):
5      name=raw_input('Δώσε όνομα υποψηφίου/ιας: ')
6      pr=int(input('Δώσε προφορική βαθμολογία υποψηφίου/ιας: '))
7      while pr<1 or pr >100:
8          pr=int(input('Δώσε σωστά προφορική βαθμολογία υποψηφίου/ιας (1-100): '))
9      gr=int(input('Δώσε γραπτή βαθμολογία υποψηφίου: '))
10     while gr<1 or gr>100:
11         gr=int(input('Δώσε σωστά γραπτή βαθμολογία υποψηφίου/ιας: '))
12     #4.2 Έλεγχος βαθμολογιών μεταξύ τους και εμφάνιση κατάλληλου μηνύματος
13     if pr>gr:
14         print 'Ο υποψήφιος/ια ', name, ' έχει μεγαλύτερη βαθμολογία στα προφορικά'
15     elif pr<gr:
16         print 'Ο υποψήφιος/ια ', name, ' έχει μεγαλύτερη βαθμολογία στα γραπτά'
17     else:
18         print 'Ο υποψήφιος/ια ', name, ' έχει ίδιες βαθμολογίες προφορικά και γραπτά'
19     #4.3 Υπολογισμός και εμφάνιση ποσοστού επιτυχόντων
20     mo=(pr+gr)/2.
21     print mo
22     if mo>=60:
23         petixe=petixe +1
24     print petixe
25     pososto=petixe/250.*100
26     print 'Στος εξετάσεις πέτυχε το ',pososto,' % των υποψηφίων'
```


ΘΕΜΑ 2

2.1. Δίνεται το παρακάτω τμήμα προγράμματος στη γλώσσα προγραμματισμού Python.

```
1   a=20
2   x=int(input('Δώσε ακέραιο αριθμό'))
3   if x<30:
4       a=a+x #εντολή 1
5   else:
6       a=a-x #εντολή 2
7   print a
```

Να γράψετε πέντε αριθμούς οι οποίοι **δίνονται στη μεταβλητή x** σε διαδοχικές εκτελέσεις. Οι δύο πρώτοι θα πρέπει να προκαλούν την εκτέλεση της εντολής 1, ο 3^{ος} και 4^{ος} να προκαλούν την εκτέλεση της εντολής 2 και με τον 5^ο (εάν υπάρχει) να μην εκτελείται ποτέ η εντολή **print a**

Μονάδες 10

2.2. Να συμπληρωθούν οι αριθμοί 1-9 έτσι ώστε το παρακάτω τμήμα προγράμματος σε Python, να **διαβάζει 10 αριθμούς**, να ελέγχει αν καθένας από αυτούς είναι μεταξύ 0 και 100 και να τυπώνει τον μεγαλύτερο και τον μικρότερο από αυτούς. Θεωρίστε ότι όλοι οι αριθμοί που εισάγονται είναι διαφορετικοί μεταξύ τους.

```
1   min= __1__
2   max=__2__
3   for i in range(__3__):
4       x=__4__(input('Δώσε τον 1ο αριθμό (0-100)'))
5       while __5__ or __6__:
6           __7__=int(input('Δώσε σωστά τον 1ο αριθμό (0-100)' ))
7       if x>max:
8           max=__8__
9       if __9__<min:
10          min=x
11  print ' μεγαλύτερος είναι ο: ',max
12  print ' Ο μικρότερος είναι ο : ',min
```

Μονάδες 15

ΘΕΜΑ 2

2.1.

Για την εντολή 1 ενδεικτικές τιμές: 29, 10

Για την εντολή 2 ενδεικτικές τιμές: 31,40

Για τον 5^ο αριθμό σημειώνεται ότι δεν υπάρχει αριθμός **για τον οποίο δεν θα εκτελεστεί η print**

2.2.

```
1  min= 100000
2  max=-1
3  for i in range(10):
4      x=int(input('Δώσε τον 1ο αριθμό (0-100)'))
5      while x<0 or x>100:
6          x=int(input('Δώσε σωστά τον 1ο αριθμό (0-100)' ))
7      if x>max:
8          max=x
9      if x<min:
10         min=x
11  print ' μεγαλύτερος είναι ο: ',max
12  print ' Ο μικρότερος είναι ο : ',min
```

ΘΕΜΑ 4

Δυο παίκτες παίζουν ένα επιτραπέζιο παιχνίδι. Στο παιχνίδι αυτό ρίχνει καθένας διαδοχικά ένα ζάρι. Κάθε ρίψη ζαριού μπορεί να είναι οποιοσδήποτε από τους ακέραιους αριθμούς από 1 έως 6. Νικητής σε κάθε γύρο του παιχνιδιού είναι όποιος φέρει μεγαλύτερο αριθμό από τον άλλον. Το παιχνίδι έχει 15 γύρους.

Να αναπτύξετε πρόγραμμα στην γλώσσα προγραμματισμού Python το οποίο:

4.1. Να διαβάζει τα ονόματα των δύο παικτών. Για **κάθε** γύρο του παιχνιδιού να διαβάζει τους αριθμούς από τις ρίψεις των δύο παικτών. Η κάθε ρίψη πρέπει να είναι αριθμός μεταξύ του 1 και του 6. Σε περίπτωση που η εισαγωγή κάποιου αριθμού είναι έξω από τα επιτρεπτά όρια να εισάγεται εκ νέου σωστός αριθμός.

Μονάδες 9

4.2. Να υπολογίζει και να εμφανίζει με κατάλληλο μήνυμα το όνομα του παίκτη, ο οποίος κέρδισε στο τέλος των 15 γύρων ή αν έχουμε ισοπαλία.

Μονάδες 9

4.3. Να υπολογίζει και να εμφανίζει με κατάλληλο μήνυμα πόσες φορές έφεραν και οι δυο παίκτες τον ίδιο αριθμό (ισοπαλία).

Μονάδες 7

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές.

Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
    print b
```

ΘΕΜΑ 4

```
1  #Αρχικοποιεί τους μετρητές για τις νίκες ή ισοπαλία των 2 παικτών
2  first=0
3  second=0
4  isopalia=0
5  #4.1 Διαβάζει τα ονόματα των παικτών
6  name1=raw_input('Δώσε όνομα πρώτου παίκτη: ')
7  name2=raw_input('Δώσε όνομα δεύτερου παίκτη: ')
8  #4.1 Διαβάζει τις ρίψεις των παικτών για 15 γύρους
9  for i in range(15):
10     a=int(input('Δώσε την ρίψη του 1ου παίκτη: '))
11     while a<1 or a>16:
12         a=int(input('Δώσε σωστά την ρίψη του 1ου παίκτη (1-6): '))
13     b=int(input('Δώσε την ρίψη του 2ου παίκτη: '))
14     while b<1 or b>16:
15         b=int(input('Δώσε σωστά την ρίψη του 2ου παίκτη (1-6): '))
16     #Υπολογίζει τις νίκες ή την ισοπαλία σε κάθε γύρο
17     if a>b:
18         first=first+1
19     elif a<b:
20         second=second+1
21     else:
22         isopalia=isopalia +1
23     # Υπολογίζει και εμφανίζει ποιός κέρδισε ή την ισοπαλία
24     if first >second:
25         print ' Κέρδισε ο παίκτης ', name1
26     elif first<second:
27         print ' Κέρδισε ο παίκτης ', name2
28     else:
29         print ' Το παιχνίδι έληξε ισόπαλο'
30     # Υπολογίζει και εμφανίζει πόσες ρίψεις είχαν ίδιο αριθμό
31     print ' Οι δύο παίκτες έφεραν τον ίδιο αριθμό ', isopalia, ' φορές'
```

ΘΕΜΑ 2

2.1. Να γράψετε συνάρτηση με όνομα `multiple` η οποία θα δέχεται μια ακέραια τιμή και θα εμφανίζει την προπαίδεια του αριθμού αυτού.

Μονάδες 13

2.2. Να γράψετε στο γραπτό σας τους αριθμούς 1, 2, 3 από τη **Στήλη Α** του παρακάτω πίνακα και δίπλα ένα από τα γράμματα **α, β, γ, δ, ε** της **Στήλης Β**, που δίνει τη σωστή αντιστοίχιση. Σημειώνεται ότι δύο γράμματα από τη **Στήλη Β** θα περισσέψουν.

Στήλη Α (τμήμα προγράμματος)	Στήλη Β (αριθμός επαναλήψεων)
1. <code>for i in range (2, 12, 4) :</code> <code>print i</code>	α. πέντε
2. <code>i = 8</code> <code>while i > -2 :</code> <code>print i</code> <code>i = i - 2</code>	β. έξι
3. <code>i = 3</code> <code>while i !=12 :</code> <code>i = i + 5</code> <code>print i</code>	γ. τρεις
	δ. Άπειρες
	ε. Τέσσερις

Μονάδες 12

ΘΕΜΑ 2

2.1

```
1  def multiple (x):  
2      print 'Η προπαίδεια του αριθμού ',x,' είναι:'  
3      for i in range (1,11):  
4          print i , ' x ',x,' = ',i*x
```

2.2

1. γ
2. α
3. δ

ΘΕΜΑ 4

Ένας ιδιωτικός μετεωρολογικός σταθμός καταγράφει καθημερινά στις 7 π.μ την θερμοκρασία του περιβάλλοντος.

Να γράψετε πρόγραμμα σε Python το οποίο:

4.1. Να διαβάζει για κάθε μέρα του Ιανουαρίου την θερμοκρασία του περιβάλλοντος ελέγχοντας οι τιμές να είναι από -10 έως 14 βαθμούς °C. Σε περίπτωση λανθασμένης τιμής να εμφανίζεται μήνυμα λάθους.

Μονάδες 8

4.2. Να υπολογίζει και εμφανίζει την μέση θερμοκρασία του μήνα.

Μονάδες 6

4.3. Να βρίσκει την χαμηλότερη θερμοκρασία του μήνα και ποια ημέρα του μήνα σημειώθηκε αυτή.

Μονάδες 7

4.4. Να υπολογίζει και εμφανίζει το πλήθος των ημερών που η θερμοκρασία ήταν κάτω από 0 βαθμούς.

Μονάδες 4

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
    print b
```

ΘΕΜΑ 4

```
#Αρχικοποίηση μεταβλητών
pl=0          #πλήθος ημέρων με θερμοκρασία <0
mikr_th=100   #μικρότερη θερμοκρασία
s=0.0         #συνολική θερμοκρασία
# 4.1 Για τις 31 ημέρες του Ιανουαρίου
for i in range (1,32):
    #4.1 Διάβασμα θερμοκρασίας
    th=float(input('Δώσε την θερμοκρασία του περιβάλλοντος την '+str(i)+'η ημέρα του Ιανουαρίου'))
    #4.1 Έλεγχος αποδεκτής τιμής θερμοκρασίας
    while th<-10 or th>14:
        print 'Έδωσες λάθος θερμοκρασία'
        th=float(input('Δώσε θερμοκρασία περιβάλλοντος στο διάστημα [-10,14]'))
    #4.2 Υπολογισμός συνολικής θερμοκρασίας
    s=s+th
    #4.3 Υπολογισμός μικρότερης θερμοκρασίας και ποια ημέρα καταγράφηκε
    if th<mikr_th:
        mikr_th=th
        min_day=i
    #4.4 Πόσες ημέρες η θερμοκρασία ήταν κάτω από 0
    if th<0:
        pl=pl+1
#4.2 Υπολογισμός και εμφάνιση μέσης θερμοκρασίας μήνα
mesi_th=s/31
print 'Η μέση θερμοκρασία του Ιανουαρίου είναι',mesi_th
#4.3 Υπολογισμός και εμφάνιση μικρότερης θερμοκρασίας και πότε καταγράφηκε
print 'Η μικρότερη θερμοκρασία καταγράφηκε στις ',min_day,'του μήνα και είναι :',mikr_th
#4.4 Εμφάνιση πόσες ημέρες η θερμοκρασία ήταν υπό το μηδέν
print pl,' ημέρες η θερμοκρασία ήταν κάτω από 0 βαθμούς'
```


ΘΕΜΑ 2

2.1 Να γράψετε τις τιμές που εμφανίζονται μετά την εκτέλεση των παρακάτω τμημάτων κώδικα σε γλώσσα προγραμματισμού Python:

α)

```
for i in range (11):
```

```
    print i,
```

β)

```
for i in range (5,40,10):
```

```
    print i,
```

γ)

```
for i in range (40,5, -10):
```

```
    print i,
```

Μονάδες 12

2.2 Να συμπληρωθούν τα αριθμημένα κενά (1-4) στο παρακάτω πρόγραμμα σε γλώσσα προγραμματισμού Python, έτσι ώστε να υλοποιεί την εξής λειτουργία:

Θα διαβάξει έναν θετικό ακέραιο. Όσο ο αριθμός είναι μεγαλύτερος της μονάδας, αν είναι άρτιος θα τον διαιρεί με το 2, ενώ αν είναι περιττός θα τον πολλαπλασιάζει με το 3 και θα προσθέτει μια μονάδα. Θα εμφανίζει τον αριθμό που προέκυψε και θα επαναλαμβάνει την παραπάνω διαδικασία για τον αριθμό αυτό.

```
x = int((1) ("Πληκτρολόγησε θετικό ακέραιο: "))
```

```
while x > 1:
```

```
    if (1) :
```

```
        x = (1)
```

```
    else:
```

```
        x = x/2
```

```
    print (1),
```

Μονάδες 13

ΘΕΜΑ 2

2.1

α) 0 1 2 3 4 5 6 7 8 9 10

β) 5 15 25 35

γ) 40 30 20 10

2.2 Το πρόγραμμα, με τα κενά συμπληρωμένα, είναι το εξής:

```
1  x = int(input("Πληκτρολόγησε θετικό ακέραιο: "))
2  while x > 1:
3      if x % 2 != 0 :
4          x = 3*x + 1
5      else:
6          x = x/2
7      print x,
```

ΘΕΜΑ 4

Σε ένα ΚΤΕΟ προσέρχονται για τεχνικό έλεγχο τριών τύπων οχήματα: Επιβατικά, Φορτηγά και Μοτοσυκλέτες. Το κόστος ελέγχου υπολογίζεται σύμφωνα με τον παρακάτω πίνακα:

Τύπος οχήματος	Κόστος ελέγχου
Επιβατικά	50 ευρώ
Φορτηγά	80 ευρώ
Μοτοσυκλέτες	30 ευρώ

Να γράψετε πρόγραμμα σε γλώσσα προγραμματισμού Python, το οποίο για μία συγκεκριμένη ημέρα:

4.1. Να διαβάζει τον τύπο του οχήματος για κάθε όχημα που προσέρχεται, «Ε» για Επιβατικό, «F» για Φορτηγό και «Μ» για Μοτοσυκλέτα. Η διαδικασία εισαγωγής δεδομένων τερματίζεται όταν δοθεί ως τύπος οχήματος η λέξη «TELOS». Απαιτείται έλεγχος εγκυρότητας όλων των δεδομένων που εισάγονται.

Μονάδες 11

4.2. Να υπολογίζει το πλήθος των οχημάτων που προσήλθαν για έλεγχο και τις εισπράξεις του ΚΤΕΟ, για κάθε τύπο οχήματος χωριστά, και να τα εμφανίζει με κατάλληλα μηνύματα.

Μονάδες 11

4.3. Να υπολογίζει και να εμφανίζει το πλήθος όλων των οχημάτων καθώς και το συνολικό ποσό είσπραξης του ΚΤΕΟ.

Μονάδες 3

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1  epiv=0 # αρχικοποιήσεις
2  fortiga=0
3  moto=0
4  se=0
5  sf=0
6  sm=0
7  # ερώτημα 4.1
8  typos=raw_input("Δώστε τύπο οχήματος ")
9  while typos!= "E" and typos!= "F" and typos!= "M" and typos!= "TELOS":
10     typos=raw_input("Δώστε αποδεκτό τύπο οχήματος ")
11
12 while typos!= "TELOS": # βρόχος επανάληψης
13
14     if typos=="E": # δομή πολλαπλήςεπιλογής
15         epiv=epiv+1 # ερώτημα 4.3
16         cost=50
17         se=se + cost # ερώτημα 4.3
18     elif typos=="F":
19         fortiga=fortiga+1 # ερώτημα 4.3
20         cost=80
21         sf=sf + cost # ερώτημα 4.3
22     else:
23         moto=moto+1 # ερώτημα 4.3
24         cost=30
25         sm=sm + cost # ερώτημα 4.3
26     typos=raw_input("Δώστε τύπο επόμενου οχήματος ")
27     while typos!= "E" and typos!= "F" and typos!= "M" and typos!= "TELOS":
28         typos=raw_input("Δώστε αποδεκτό τύπο οχήματος ")
29 # Ερώτημα 4.2
30 print "Επιβατικά ", epiv, se # εμφάνιση αποτελεσμάτων
31 print "Φορτηγά ", fortiga, sf
32 print "Μοτοσυκλέτες ", moto, sm
33 # Ερώτημα 4.3
34 print "Πλήθος όλων ", epiv + fortiga + moto
35 print "Συνολικό ποσό είσπραξης ", se+sf+sm
```

ΘΕΜΑ 2

2.1 Να συμπληρώσετε τα αριθμημένα κενά (1-5) στο παρακάτω πρόγραμμα σε γλώσσα προγραμματισμού Python το οποίο ζητά μια λέξη από τον χρήστη και εμφανίζει πόσα φωνήεντα (χωρίς τόνους) περιέχει:

```
word = (1) ("Δώστε μια ελληνική λέξη χωρίς τόνους")
```

```
metritis = (2)
```

```
for c in word:
```

```
    if (3) in "αηιουωΑΕΗΙΟΘΩ":
```

```
        metritis = (4) + 1
```

```
print "Η λέξη περιέχει ", (5) , " φωνήεντα"
```

Μονάδες 10

2.2 Να γράψετε στο τετράδιό σας τους αριθμούς 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 από τη Στήλη Α του παρακάτω πίνακα και δίπλα ένα από τα γράμματα α, β, γ, δ, ε, στ, ζ, η, θ, ι, κ της Στήλης Β, που δίνει τη σωστή αντιστοιχία. Σημειώνεται ότι ένα γράμμα θα περισσέψει.

ΣΤΗΛΗ Α	ΣΤΗΛΗ Β
1. if	α. Λογικός τελεστής
2. False	β. Συγκριτικός τελεστής
3. "True"	γ. Λογική τιμή
4. or	δ. Συμβολοσειρά
5. ==	ε. Αριθμητικός τελεστής
6. *	στ. Εντολή επιλογής
7. for	ζ. Εκχώρηση τιμών
8. =	η. Εντολή επανάληψης
9. print	θ. Εντολή εισόδου τιμών
10. input	ι. Εντολή εξόδου
	κ. Πλειάδα

Μονάδες 15

ΘΕΜΑ 2

2.1.

(1): raw_input

(2): 0

(3): c

(4): metritis

(5): metritis

2.2

1.στ 2.γ 3.δ 4.α 5.β 6.ε 7.η 8.ζ 9.ι 10.θ

ΘΕΜΑ 4

Σε μια εξέταση του μαθήματος Προγραμματισμός Υπολογιστών εξετάζονται εκατό (100) μαθητές προφορικά και γραπτά. Οι μαθητές βαθμολογούνται από το 0.0 έως και το 20.0 σε κάθε εξέταση (προφορική και γραπτή). Να γράψετε ένα πρόγραμμα σε γλώσσα προγραμματισμού Python το οποίο να υλοποιεί τα παρακάτω:

4.1 Να διαβάξει το ονοματεπώνυμο, την προφορική και τη γραπτή βαθμολογία κάθε μαθητή. Απαιτείται έλεγχος ορθότητας της προφορικής και γραπτής βαθμολογίας που εισάγονται.

Μονάδες 7

4.2 Να εμφανίζει τα ονοματεπώνυμα των μαθητών που έχουν μέσο όρο προφορικής και γραπτής βαθμολογίας μεγαλύτερο από το 18.

Μονάδες 9

4.3 Να υπολογίζει και να εμφανίζει το πλήθος των μαθητών που η γραπτή βαθμολογία τους είναι μικρότερη από την προφορική τους.

Μονάδες 9

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```


ΘΕΜΑ 4

```
1  pl = 0 # Αρχικοποίηση
2  for i in range(100): # Βρόχος επανάληψης
3      name = raw_input("Δώσε ονοματεπώνυμο ") # 4.1
4      pr = float(input("Δώσε προφορικό βαθμό "))
5      while pr<0.0 or pr>20.0 : # έλεγχος εγκυρότητας
6          pr = float(input("Δώσε προφορικό βαθμό "))
7      gr = float(input("Δώσε γραπτό βαθμό "))
8
9      while gr<0.0 or gr>20.0 :
10         gr = float(input("Δώσε γραπτό βαθμό "))
11         mo = (pr+gr)/2
12
13         if mo > 18 : # 4.2
14             print name
15         if gr < pr : # 4.3
16             pl = pl+1 # Ενημέρωση μετρητή
17     print "Πλήθος ατόμων με γραπτό μικρότερο του προφορικού: ", pl
```

ΘΕΜΑ 2

2.1 Να γράψετε κώδικα σε Python με τον οποίο επεξεργάζεστε μία λίστα ως εξής:

- α) Εισάγετε στη λίστα τους αριθμούς 50, 35, 75, 68, 985 και 156.
- β) Τυπώστε το πέμπτο στοιχείο της λίστας.
- γ) Εκχωρήστε ως νέα τιμή του τελευταίου στοιχείου της λίστας το 10.
- δ) Προσθέστε στο τέλος της λίστας τον αριθμό 99.
- ε) Τυπώστε το πλήθος των στοιχείων της λίστας.

Μονάδες 10

2.2 Να ξαναγράψετε το παρακάτω τμήμα προγράμματος σε Python, χρησιμοποιώντας την εντολή επανάληψης while αντί της εντολής for έτσι ώστε να εμφανίζει το ίδιο αποτέλεσμα.

```
sum=0
```

```
for i in range(1,20,2):
```

```
    sum=sum+i
```

```
    print sum
```

```
print sum
```

Μονάδες 15

ΘΕΜΑ 2

2.1

α) lista = [50, 35, 75, 68, 985, 156]
β) print "Η πέμπτη τιμή της λίστας είναι ", lista[4]
γ) lista[5] = 10
δ) lista.append(99)
ε) print "Το πλήθος των στοιχείων είναι ", len(lista)

2.2

```
sum=0
i=1
while i<20:
    sum=sum + i
    print sum
    i=i+2
print sum
```

ΘΕΜΑ 4

Στο πλαίσιο της ορθής κατανομής των μαθητών/τριών στις σχολικές μονάδες, να γράψετε πρόγραμμα σε γλώσσα προγραμματισμού Python το οποίο να πραγματοποιεί τα παρακάτω:

4.1 Να διαβάζει, με χρήση μιας δομής επανάληψης, το όνομα κάθε σχολείου και το πλήθος των μαθητών/τριών του. Η διαδικασία αυτή τερματίζεται όταν δοθεί ως όνομα σχολείου η λέξη «TELOS» (θεωρείστε ότι συμμετέχουν τουλάχιστον 2 σχολεία). Να γίνεται έλεγχος εγκυρότητας εισαγωγής δεδομένων έτσι ώστε το πλήθος των μαθητών κάθε σχολείου να είναι μεγαλύτερο του 0 και μικρότερο του 500.

Μονάδες 10

4.2 Να εμφανίζει το όνομα του σχολείου και το πλήθος των μαθητών για το σχολείο που έχει τους περισσότερους και για το σχολείο που έχει τους λιγότερους μαθητές (θεωρείστε ότι ο αριθμός των μαθητών είναι διαφορετικός για κάθε σχολείο).

Μονάδες 10

4.3 Να υπολογίζει και να εμφανίζει με κατάλληλο μήνυμα τον μέσο όρο του αριθμού των μαθητών όλων των σχολείων.

Μονάδες 5

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
    print b
```

ΘΕΜΑ 4

```
1  pl = 0 # αρχικοποιήσεις
2  s = 0
3  minm = 500
4  maxm = 0
5  # 4.1
6  sxol = raw_input("Δώστε όνομα σχολείου ")
7  while sxol != "TELOS" : # βρόχος επανάληψης
8      e = input("Δώστε αριθμό μαθητών/τριών ")
9      while e <1 or e > 499 : # έλεγχος εγκυρότητας
10         e = input("Δώστε αριθμό μαθητών/τριών ")
11     pl = pl+1
12     s = s+e
13     if e > maxm : # 4.2
14         maxm = e
15         smax = sxol
16     if e < minm :
17         minm = e
18         smin = sxol
19     sxol = raw_input("Δώστε όνομα σχολείου ")
20
21 print "Σχολείο με τους περισσότερους μαθητές/τριες ", smax, maxm
22 print "Σχολείο με τους λιγότερους μαθητές/τριες ", smin, minm
23 mo = s / pl # 4.3
24 print "Ο μέσος όρος των μαθητών είναι: ", mo
```

ΘΕΜΑ 2

2.1. Στην παρακάτω εντολή σε γλώσσα προγραμματισμού Python, οι μεταβλητές a, b και c είναι τύπου integer και έχουν τις ακόλουθες τιμές: a = -4, b = 2 και c = 4

x = (abs(a) == b+c -2) and (28 == 2 * pow(b, 4) - 4)

Να γράψετε την τιμή που θα πάρει η μεταβλητή x όταν εκτελεστεί η εντολή αυτή

Μονάδες 10

2.2 Στο παρακάτω τμήμα προγράμματος σε Python υπάρχουν πέντε (5) κενά, τα οποία έχουν αριθμηθεί και υπογραμμιστεί. Να γράψετε στο γραπτό σας τον αριθμό του κάθε κενού και δίπλα τι πρέπει να συμπληρωθεί, ώστε το τμήμα προγράμματος να εμφανίζει αντεστραμμένες τις τιμές του x και y, δηλαδή 10 και 5 αντίστοιχα.

```
1  x = 5
2  y = 10
3
4  print x, y
5  (1) = x + y
6  y = x - (2)
7  x = (3) - y
8  print (4) , (5)
```

Μονάδες 15

ΘΕΜΑ 2

2.1

True

2.2

1. x
2. y
3. x
4. x
5. y

ΘΕΜΑ 2

2.1. Δίνεται το παρακάτω πρόγραμμα σε γλώσσα Python:

```
1  n = int(input("Δώσε ένα μονοψήφιο ακέραιο:"))
2  result=0
3  if n > 1:
4      if n%2==0:
5          print n
6      else:
7          print n
8  else:
9      n=1
10 result = (n-1) + n
11 print 'Αποτέλεσμα: ', result
```

A. Να γράψετε στο γραπτό σας ποιες τιμές θα εμφανιστούν στην οθόνη κατά την εκτέλεση των εντολών των σειρών 5 και 7 του προγράμματος αν δώσουμε τις τιμές -1, 0, 1, 2, 3

Μονάδες 10

B. Να γράψετε στο γραπτό σας το τελικό αποτέλεσμα από την εκτέλεση του προγράμματος με τις παραπάνω τιμές, δηλ. -1, 0, 1, 2, 3

Μονάδες 5

2.2 Σημειώστε στο γραπτό σας ποια θα είναι η τελική μορφή της λίστας R που θα εμφανιστεί στην οθόνη μετά την εφαρμογή των παρακάτω εντολών:

```
1  R=[50,40,30,20,10]
2  R[0]= R[0]/5
3  R[2]= R[1]-R[2]
4  R[4]= R[0]* R[0]/ R[0]
5  R[2]= R[1]- R[0]
6  R[0]= pow(R[4],2)- R[2]-R[3]
7  print R
```

Μονάδες 10

ΘΕΜΑ 2

2.1

A.

Τιμή	Εμφάνιση τιμών
-1	{Κενό}
0	{Κενό}
1	{Κενό}
2	2
3	3

B.

Τιμή	Αποτέλεσμα
-1	1
0	1
1	1
2	3
3	5

2.2

[50, 40, 30, 20, 10]

ΘΕΜΑ 2

2.1. Δίνεται το παρακάτω τμήμα κώδικα σε γλώσσα προγραμματισμού Python:

```
1 text = "επιπέδου"  
2  
3 def myMoto(text,nr):  
4     text = "A"*3  
5     return text  
6  
7 message= "Η Python είναι γλώσσα ποιότητας " + myMoto(text,3)  
8 print message  
9 print "Η Python είναι μια γλώσσα υψηλού " + text
```

Να γράψετε στο γραπτό σας τι θα εμφανιστεί στην οθόνη κατά την εκτέλεση των εντολών του παραπάνω κώδικα.

Μονάδες 10

2.2

Στο παρακάτω τμήμα προγράμματος σε Python υπάρχουν πέντε (5) κενά, τα οποία έχουν αριθμηθεί και υπογραμμιστεί. Να γράψετε στο γραπτό σας τον αριθμό του κάθε κενού και δίπλα τι πρέπει να συμπληρωθεί, ώστε το τμήμα προγράμματος να διαβάζει ένα ποσό ανάληψης (πολλαπλάσιο του 10) και να υπολογίζει το πλήθος χαρτονομισμάτων των 50, 20 και 10 ευρώ που αντιστοιχούν σε αυτό.

```
1 poso = int(input('Δώσε ποσό ανάληψης: '))  
2 (1) = poso / 50  
3 rest = poso % (2)  
4 r20 = (3) / 20  
5 rest = rest % (4)  
6 r10 = rest / 10  
7 print r50, r20, (5)
```

Μονάδες 15

ΘΕΜΑ 2

2.1

Η Python είναι γλώσσα ποιότητας AAA

Η Python είναι μια γλώσσα υψηλού επιπέδου

2.2

1. r50
2. 50
3. rest
4. 20
5. r10

ΘΕΜΑ 2

2.1. Το παρακάτω πρόγραμμα σε γλώσσα προγραμματισμού Python δημιουργήθηκε για να ελέγχει τα αποτελέσματα 13 αγώνων του ΠΡΟΠΟ που εισάγονται από το πληκτρολόγιο (Τιμές: 1, 2 ή X). Μετράει και εμφανίζει το πλήθος νικών των γηπεδούχων (1), νικών των φιλοξενούμενων (2) και ισοπαλιών (X). Υπάρχουν όμως πέντε (5) λάθη. Γράψτε στο γραπτό σας τον αριθμό της σειράς και δίπλα το λάθος.

```
1   agones= 14
2   m1=0
3   mx=x
4   m2=0
5   for x in range(0,agones):
6       apot = int(raw_input('Παρακαλώ δώστε 1, 2 ή X: '))
7       if apot=='1':
8           m1=m1+1
9       elif apot=='2':
10          m2=m2+2
11       else:
12          mx=mx+x
13       print m1, m2, mx
```

Μονάδες 10

2.2.

A. Σημειώστε στο γραπτό σας το γράμμα όσων από τις παρακάτω εντολές είναι έγκυρες στη γλώσσα προγραμματισμού Python:

A. integer = 4%2

B. class 2= "B' ΕΠΑΛ"

Γ. s1, s2 = 0.24 , 1.24

Δ. value ← 19.95

Μονάδες 5

B. Για όσες από τις παραπάνω εντολές είναι λανθασμένες, να περιγράψετε το λάθος τους.

Μονάδες 10

ΘΕΜΑ 2

2.1

Σειρά	Λάθος
1.	13
3.	mx=0
6.	Int()
10.	1
12.	1

2.2

A.

A

Γ

B.

Η εντολή B (class 2= "B' ΕΠΑΛ") είναι λανθασμένη γιατί στην Python δεν επιτρέπεται τα ονόματα των μεταβλητών να έχουν κενό. Αντί κενού πρέπει να χρησιμοποιείται _.

Η εντολή Δ (value ← 19.95) είναι λανθασμένη γιατί στην Python η εκχώρηση τιμής σε μια μεταβλητή γίνεται με το ίσον (=) κι όχι με το βελάκι (←). (σελ. 45 σχολικού βιβλίου)

ΘΕΜΑ 2

2.1. Να σημειώσετε στο γραπτό σας το γράμμα και δίπλα τη λίστα αριθμών που παράγει η καθεμία από τις παρακάτω συναρτήσεις range. Σε περίπτωση που κάποια range δεν παράγει λίστα με αριθμούς τότε γράψτε [].

- A. range (-5)
- B. range(-5,0,5)
- Γ. range(5, -1, 1)
- Δ. range(5, 101, 5)
- E. range(5, 105, 5)

Μονάδες 10

2.2 Στο παρακάτω τμήμα προγράμματος σε Python υπάρχουν πέντε (5) κενά, τα οποία έχουν αριθμηθεί και υπογραμμιστεί. Να σημειώσετε στο γραπτό σας τον αριθμό του κάθε κενού και δίπλα τι πρέπει να συμπληρωθεί, ώστε το τμήμα προγράμματος να συγκρίνει και να εμφανίσει την φιάλη κρασιού με την πιο συμφέρουσα τιμή, δηλαδή την τρίτη.

```
1  val_per_ml_1 = 15.80
2  val_per_ml_2 = 18.30
3  val_per_ml_3 = 12.65
4  if (1) < val_per_ml_2 :
5      if val_per_ml_1 (2) val_per_ml_3:
6          print "No1 Κρήτης νάμα"
7      elif val_per_ml_1 >= (3) :
8          print "No3 Κτήμα Πελοποννήσου"
9      elif val_per_ml_1 >= (4) :
10     if val_per_ml_2 < val_per_ml_3:
11         print "No2 Γεύση Μακεδονίας"
12     elif val_per_ml_2 (5) val_per_ml_3:
13         print "No3 Κτήμα Πελοποννήσου"
```

Μονάδες 15

ΘΕΜΑ 2

2.1

A. []

B. [-5]

Γ. []

Δ. [5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80, 85, 90, 95, 100]

Ε. [5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80, 85, 90, 95, 100]

2.2

1. val_per_ml_1
2. <
3. val_per_ml_3
4. val_per_ml_2
5. >=

ΘΕΜΑ 4

Στον ετήσιο αγώνα ταχύτητας καρτ συμμετέχουν 12 έφηβοι οδηγοί. Ο κάθε οδηγός οδηγεί μόνος του και χρονομετρείται. Οι χρόνοι κρατούνται σε λεπτά (ακέραιος) και υποτίθεται ότι όλοι οι οδηγοί ολοκληρώνουν τη διαδρομή με διαφορετικό χρόνο. Να αναπτυχθεί πρόγραμμα σε γλώσσα προγραμματισμού Python, το οποίο:

4.1. Να διαβάζει το όνομα των κάθε οδηγού.

Μονάδες 3

4.2. Να υπολογίζει και να εμφανίζει, με κατάλληλο μήνυμα, το όνομα και τον χρόνο του ταχύτερου οδηγού δηλαδή αυτού με τον μικρότερο χρόνο όλου του αγώνα.

Μονάδες 7

4.3. Να υπολογίζει και εμφανίζει, με κατάλληλο μήνυμα, το όνομα και τον χρόνο του οδηγού με τον μεγαλύτερο χρόνο σε ένα γύρο του αγώνα.

Μονάδες 7

4.4. Να υπολογίζει και να εμφανίζει, με κατάλληλο μήνυμα, το μέσο χρόνο όλων των οδηγών του αγώνα.

Μονάδες 8

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```


ΘΕΜΑ 4

```
drive_nr=12      # πλήθος οδηγών
min_time= 999999 # αρχικοποίηση μετρητή ελάχιστου χρόνου
max_time = -1    # αρχικοποίηση μετρητή μέγιστου χρόνου
tot_time= 0      # αρχικοποίηση μετρητή συνολικού χρόνου
for i in range(drive_nr):
    drive_name=raw_input('Δώσε ονοματεπώνυμο οδηγού: ') #4.1 εισαγωγή ονοματεπώνυμου
    drive_time=int(input('Δώσε χρόνο αγώνα: ')) #4.1 εισαγωγή χρόνου αγώνα
    tot_time= tot_time+drive_time # συνολικός χρόνος οδηγών
    if drive_time < min_time:
        min_time = drive_time #4.2 μικρότερος χρόνος αγώνα
        fast_round_name = drive_name #4.2 ονοματεπώνυμο οδηγού μικρότερου χρόνου

    if drive_time > max_time:
        max_time = drive_time #4.3 μεγαλύτερο χρόνος ενός αγώνα
        slow_round_name = drive_name #4.3 ονοματεπώνυμο οδηγού μεγαλύτερου χρόνου

    if drive_time < min_time:
        min_time = drive_time #4.3 μικρότερος χρόνος όλου του αγώνα
        drive_name = drive_name #4.3 ονοματεπώνυμο οδηγού
#4.2 ταχύτερος οδηγός αγώνα
print 'Ταχύτερος οδηγός αγώνα:', fast_round_name, 'με χρόνο: ', min_time
#4.4 Βραδύτερος οδηγός αγώνα
print 'Βραδύτερος οδηγός αγώνα:', slow_round_name, 'με χρόνο: ', max_time
mesos_xronos= tot_time/drive_nr #4.4 μέσος χρόνος οδηγών αγώνα
print 'Ο μέσος χρόνος αγώνα είναι:', mesos_xronos
```

ΘΕΜΑ 4

Να αναπτυχθεί πρόγραμμα σε γλώσσα προγραμματισμού Python, το οποίο

4.1 Να εισάγονται από το πληκτρολόγιο 7 τυχαίοι αριθμοί από το 10 μέχρι και το 99 που όλοι τους θα είναι διαφορετικοί. Να γίνεται έλεγχος ορθότητας ώστε να εισάγονται μόνο αποδεκτές τιμές από το 10 μέχρι και το 99.

Μονάδες 5

4.2 Να εντοπίζει τον μικρότερο και τον μεγαλύτερο αριθμό και να τους εμφανίζει με σχετικά μηνύματα.

Μονάδες 8

4.3. Να υπολογίζει και να εμφανίζει το άθροισμα του μικρότερου και του μεγαλύτερου αριθμού. Αν αυτό είναι άρτιο τότε να εμφανίζει όλους τους αριθμούς (ένα προς ένα) στο διάστημα μεταξύ του μικρότερου και του μεγαλύτερου αριθμού με αύξουσα αρίθμηση διαφορετικά αν είναι περιττό τότε να εμφανίζει όλους τους αριθμούς (ένα προς ένα) στο διάστημα μεταξύ του μεγαλύτερου και του μικρότερου αριθμού με φθίνουσα αρίθμηση.

Μονάδες 12

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές.

Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1  pos = 7      # αρχικοποίηση μεταβλητής πλήθους αριθμών
2  elaxistos=999 # αρχικοποίηση μεταβλητής μικρότερης τιμής
3  megistos = -1 # αρχικοποίηση μεταβλητής μεγαλύτερης τιμής
4
5  for i in range(pos): # 4.1 εισαγωγή 7 διψήφιων
6      numero=int(input('Δώσε ένα ακέραιο από 10 έως και 99: '))#4.1 εισαγωγή
    αριθμού
7      while (numero<10) or (numero>99):    # 4.1 έλεγχος εγκυρότητας αριθμού
8          print 'Αποδεκτή τιμή μεταξύ 10-99'
9          numero=int(input('Δώσε ένα ακέραιο από 10 έως και 99: ')) #4.2
    επαναεισαγωγή
10     if numero < elaxistos:
11         elaxistos = numero    #4.2 μικρότερος αριθμός
12     if numero > megistos:
13         megistos = numero    #4.2 μεγαλύτερος αριθμός
14
15     print 'Ο μικρότερος αριθμός είναι: ', elaxistos
16     print 'Ο μεγαλύτερος αριθμός είναι: ', megistos
17     synolo = elaxistos+megistos    # 4.3 άθροισμα μικρότερου και μεγαλύτερου
18     print 'Άθροισμα ελάχιστου και μέγιστου: ', synolo
19     if (synolo % 2) == 0:
20         for i in range(elaxistos,megistos+1): # 4.3 εμφάνιση με αύξουσα αρίθμηση
21             print i
22     else:
23         for i in range(megistos,elaxistos-1,-1): # 4.3 εμφάνιση με φθίνουσα αρίθμηση
24             print i
```

ΘΕΜΑ 4

Η μαρίνα ενός μικρού αλλά δημοφιλούς νησιού της Ελλάδας διαθέτει 30 χώρους για στάθμευση και ανεφοδιασμό σκαφών αναψυχής. Την 15^η Αυγούστου πρόσδεσαν 30 σκάφη αναψυχής τα οποία ανεφοδιάστηκαν σε καύσιμα και απέπλευσαν την επόμενη μέρα. Να αναπτυχθεί πρόγραμμα σε γλώσσα προγραμματισμού Python, το οποίο:

4.1. Να διαβάζει το όνομα του κάθε σκάφους, το μήκος του σε μέτρα (ακέραιο) καθώς και τα λίτρα καυσίμων που ανεφοδιάστηκε. Να γίνεται έλεγχος ορθότητας ώστε να μην εισάγονται αρνητικές τιμές ή 0 για το μήκος και τα λίτρα καυσίμων των σκαφών.

Μονάδες 12

4.2. Να υπολογίζει και να εμφανίζει, με κατάλληλο μήνυμα, το όνομα και το μήκος του σκάφους με το μεγαλύτερο μήκος που πρόσδεσε.

Μονάδες 8

4.3. Να υπολογίζει και να εμφανίζει, με κατάλληλο μήνυμα, το μέσο όρο καυσίμων όλων των σκαφών.

Μονάδες 5

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές.

Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1  ship_nr=30    # πλήθος σκαφών
2  max_length=-1 # μετρητής μέγιστου μήκους
3  sum_ltr=0     # αθροιστής καυσίμων
4  for i in range(ship_nr):
5      ship_name=raw_input('Δώσε ονομασία σκάφους: ') #4.1 εισαγωγή ονομασία σκάφους
6      ship_mikos=int(input('Δώσε μήκος σκάφους: ')) #4.1 εισαγωγή μήκος σκάφους
7      while (ship_mikos<1):      # 4.1 έλεγχος εγκυρότητας εισαγόμενου μήκους
8          print 'Αποδεκτές τιμές: θετικοί ακέραιοι '
9          ship_mikos=int(input('Ξαναδώσε μήκος σκάφους: '))# 4.1 εισαγωγή μήκους σκάφους
10     ship_ltr=int(input('Δώσε λίτρα καυσίμων: ')) #4.1 εισαγωγή λίτρα καυσίμων
11     while (ship_ltr<1):      # 4.1 έλεγχος εγκυρότητας εισαγόμενων λίτρων
12         print 'Αποδεκτές τιμές: θετικοί ακέραιοι '
13         ship_ltr=int(input('Ξαναδώσε λίτρα καυσίμων: ')) # 4.1 εισαγωγή λίτρα καυσίμων
14     if ship_mikos > max_length:
15         max_length = ship_mikos    #4.2 μεγαλύτερο μήκος σκάφους
16         max_name = ship_name      #4.2 ονομασία σκάφους με το μεγαλύτερο μήκος
17         sum_ltr = sum_ltr + ship_ltr #4.3 άθροισμα καυσίμων των σκαφών
18     print 'Το σκάφος: ', max_name, ' έχει το μεγαλύτερο μήκος με : ', max_length, ' μέτρα'
19     meso_ltr = int(sum_ltr /ship_nr) #4.3 μέσος όρος καυσίμων των σκαφών
20     print 'Ο μέσος όρος καυσίμων των σκαφών είναι: ', meso_ltr, ' λίτρα'
```

ΘΕΜΑ 4

Το υπουργείο Εξωτερικών διοργανώνει εξέταση γνώσης της Αγγλικής και Γαλλικής γλώσσας προκειμένου να επιλέξει από τους 120 υποψήφιους αυτούς που θα προχωρήσουν στο επόμενο στάδιο της διαδικασίας πρόσληψης. Να αναπτυχθεί πρόγραμμα σε γλώσσα προγραμματισμού Python, το οποίο:

4.1. Να διαβάζει τον βαθμό του κάθε υποψήφιου στην Αγγλική και στην Γαλλική γλώσσα. Να γίνεται έλεγχος ορθότητας ώστε να εισάγονται μόνο αποδεκτές τιμές από 1 μέχρι και 100.

Μονάδες 8

4.2. Εφόσον οι βαθμοί του υποψήφιου στα Αγγλικά και στα Γαλλικά είναι από 80 και πάνω να διαβάζει και να καταχωρεί το ονοματεπώνυμο του σε μια λίστα με όνομα `passed`.

Μονάδες 5

4.3. Να υπολογίζει και να εμφανίζει, με κατάλληλο μήνυμα, το πλήθος των επιτυχόντων υποψηφίων.

Μονάδες 5

4.4. Να εμφανίζει το ονοματεπώνυμο όλων των καταχωρηθέντων υποψηφίων στη λίστα `pass`.

Μονάδες 7

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές.

Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1  passed=[]      # αρχικοποίηση λίστας επιτυχόντων
2  plithos=0      # αρχικοποίηση μεταβλητής πλήθους επιτυχόντων
3  cand = 20      # αρχικοποίηση μεταβλητής υποψηφίων
4  for i in range(cand):
5      eng=int(input('Δώσε βαθμό αγγλικής από 1 έως και 100: '))#4.1 εισαγωγή βαθμού
6      while (eng<1) or (eng>100):    # 4.1 έλεγχος εγκυρότητας βαθμού
7          print 'Αποδεκτός βαθμός από 1 έως και 100'
8          eng=int(input('Δώσε βαθμό αγγλικής από 1 έως και 100: ')) #4.2 επανεισαγωγή
9      fra=int(input('Δώσε βαθμό γαλλικής από 1 έως και 100: '))#4.1 εισαγωγή βαθμού
10     while (fra<1) or (fra>100):    # 4.1 έλεγχος εγκυρότητας βαθμού
11         print 'Αποδεκτός βαθμός από 1 έως και 100'
12         fra=int(input('Δώσε βαθμό γαλλικής από 1 έως και 100: ')) #4.2 επανεισαγωγή
13     if eng >=80 and fra>=80:
14         onoma= raw_input('Δώσε ονοματεπώνυμο: ') #4.2 εισαγωγή ονοματεπώνυμου επιτυχόντος
15         passed.append(onoma)
16         plithos=plithos+1    # 4.2. αύξηση μετρητή επιτυχόντων
17     print 'To πλήθος επιτυχόντων είναι: ', plithos
18     for i in range(0,plithos): # 4.4 εμφάνιση ονοματεπώνυμου λίστας επιτυχόντων
19         print passed[i]
```

ΘΕΜΑ 4

Στους προκριματικούς αγώνες για το ευρωπαϊκό πρωτάθλημα καλλιτεχνικού πατινάζ συμμετέχουν δώδεκα (12) αθλήτριες που βαθμολογούνται από 3 κριτές. Η κλίμακα βαθμολόγησης είναι από 1 μέχρι και 10. Οι αθλήτριες προκρίνονται εφόσον συγκεντρώσουν συνολικά από 25 και πάνω και κανένας κριτής δεν τις έχει βαθμολογήσει κάτω από 7. Να αναπτυχθεί πρόγραμμα σε γλώσσα προγραμματισμού Python, το οποίο:

4.1. Να διαβάζει το ονοματεπώνυμο των δώδεκα αθλητριών και τις βαθμολογίες των τριών κριτών τους. Να εκτελείται έλεγχος εγκυρότητας για αποδεκτές τιμές βαθμολογίας από 1 μέχρι και 10. Στη περίπτωση που εισάγεται άκυρη τιμή να εμφανίζεται το μήνυμα «Βαθμός εκτός ορίων. Επιτρεπτές τιμές: 1-10» και να ζητείται νέα τιμή μέχρι να δοθεί κάποια έγκυρη.

Μονάδες 12

4.2. Να ελέγχει για κάθε αθλήτρια αν προκρίνεται με βάση τις δύο προϋποθέσεις πρόκρισης και να εμφανίζει κατάλληλο μήνυμα π.χ. « Η αθλήτρια: » Ονοματεπώνυμο «προκρίνεται με βαθμό» Χ διαφορετικά αν δεν προκρίνεται να εμφανίζει το μήνυμα « Η αθλήτρια: » Ονοματεπώνυμο « δεν προκρίνεται».

Μονάδες 8

4.3. Να υπολογίζει και να εμφανίζει, με κατάλληλο μήνυμα, το πλήθος των μη προκριθέντων αθλητριών.

Μονάδες 5

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```


ΘΕΜΑ 4

```
1  scate_nr=12  # πλήθος αθλητριών
2  metr=0      # αρχικοποίηση μεταβλητής μέτρησης μη προκριθέντων αθλητριών
3  for i in range(scate_nr):
4      sc_name=raw_input('Δώσε ονοματεπώνυμο αθλήτριας: ') #4.1 εισαγωγή ονοματεπώνυμου
5      judge=3
6      tot_points =0  # μηδενισμός μετρητή συνόλου βαθμών αθλήτριας
7      for j in range (judge):
8          print 'Εισαγωγή βαθμού κριτή: ', j+1
9          j_point=int(input('Δώσε βαθμό: '))  #4.1 εισαγωγή βαθμού κριτή
10         while (j_point<1) or (j_point>10):  # 4.1 έλεγχος εγκυρότητας βαθμού
11             print 'Τιμή εκτός ορίων. Επιτρεπτά όριο 1-10'
12             j_point=int(input('Ξαναδώσε βαθμό: ')) #4.1 επανεισαγωγή βαθμού κριτή
13         if j_point >= 7:  # 4.2 έλεγχος βαθμού κάτω από 7
14             tot_points= tot_points+j_point  # 4.2 άθροιση βαθμών αθλήτριας
15         else:
16             tot_points=0  #4.2 μηδενισμός και αποκλεισμός αθλήτριας
17         if tot_points >= 25:
18             print 'Η αθλήτρια: ', sc_name, ' προκρίνεται με βαθμό: ', tot_points
19         else:
20             print 'Η αθλήτρια: ', sc_name, ' δεν προκρίνεται'
21         metr = metr +1  #4.3 μετρητής μη προκριθέντων αθλητριών
22     print '\n'
23     print 'Πλήθος αποκλεισθέντων αθλητριών:', metr #4.3 Εκτύπωση πλήθους αποκλεισθέντων
```

ΘΕΜΑ 2

2.1. Δίνεται το παρακάτω πρόγραμμα σε γλώσσα προγραμματισμού Python:

```
1 a=input("Δώσε προφορικό 1ου τετραμήνου:")
2 b=input("Δώσε προφορικό 2ου τετραμήνου:")
3 mo=(a+b)/2
4 print "Ο Μέσος όρος προφορικής βαθμολογίας είναι: ",mo
5 c=input("Δώσε το βαθμό της γραπτής εξέτασης:")
6 mo_final=(mo+c)/2
7 print "Μέσος όρος μαθήματος:",mo_final
```

Να γράψετε ισοδύναμο πρόγραμμα σε γλώσσα προγραμματισμού Python, όπου πρώτα ο μέσος όρος της προφορικής βαθμολογίας και έπειτα ο μέσος όρος μαθήματος να υπολογίζονται καλώντας την συνάρτηση `mesos_oros(a,b)`, η οποία υπολογίζει και επιστρέφει στο κυρίως πρόγραμμα τον μέσο όρο των τιμών `a` και `b`.

Μονάδες 15

2.2. Δίνεται το παρακάτω πρόγραμμα σε γλώσσα Python:

```
1 a=[1,2,3,4,5]
2 b=a[:]
3 j=4
4 x=int(j/2)
5 for i in range(0,x):
6     a[i],a[j]=a[j],a[i]
7     j=j-1
8 print "Λίστα a:",a
9 print "Λίστα b:",b
```

Να γράψετε στο γραπτό σας τι εμφανίζει το παραπάνω πρόγραμμα κατά την εκτέλεσή του.

Μονάδες 10

ΘΕΜΑ 2

2.1

A.

```
1 a=input("Δώσε προφορικό 1ου τετραμήνου:")
2 b=input("Δώσε προφορικό 2ου τετραμήνου:")
3 x=mesos_oros(a,b) # Κάλεσμα συνάρτησης
4 print "Ο Μέσος όρος προφορικής βαθμολογίας είναι: ",x
5 c=input("Δώσε το βαθμό της γραπτής εξέτασης:")
6 mo_final=mesos_oros(x,c) #Κάλεσμα συνάρτησης
7 print "Μέσος όρος μαθήματος:",mo_final
```

2.2.

Λίστα a: [5, 4, 3, 2, 1]

Λίστα b: [1, 2, 3, 4, 5]

ΘΕΜΑ 2

2.1 Να γράψετε στο γραπτό σας τα αποτελέσματα των λογικών εκφράσεων (False ή True)

αν $x=4$ και $y=2$:

α. $\text{not}(x < y)$

β. $(x > 7) \text{ and } (y < 3)$

γ. $(x \neq 10) \text{ and } (y \neq 5)$

δ. $(x > y) \text{ or } (x ** 2 < y)$

Μονάδες 8

2.2 α) Να ξαναγράψετε το παρακάτω τμήμα προγράμματος σε Python, χρησιμοποιώντας την εντολή επανάληψης `while` αντί της εντολής επανάληψης `for` έτσι ώστε να εμφανίζει στην οθόνη τα ίδια αποτελέσματα.

```
1 sum=5
2 for j in range(3,20,4):
3     sum=sum+j
4     print sum
5 print sum
```

β) Επιπλέον να γράψετε και τα κοινά αποτελέσματα που εμφανίζονται στην οθόνη.

Μονάδες 17

ΘΕΜΑ 2

2.1

α. True

β. False

γ. True

δ. True

2.2

α)

```
1 sum=5
2 i=3
3 while j<20:
4     sum=sum + j
5     print sum
6     j=j+4
7 print sum
```

β)

8

15

26

41

60

60

ΘΕΜΑ 4

Σε έναν μαθητικό διαγωνισμό στίβου, στο αγώνισμα του τριπλούν συμμετέχουν στα προκριματικά 30 μαθητές από όλα τα σχολεία ενός Δήμου. Στον τελικό περνούν όσοι μαθητές σημειώνουν επίδοση μεγαλύτερη των 7,5 μέτρων. Κάθε μαθητής έχει δύο προσπάθειες για να προκριθεί. Αν στην πρώτη προσπάθεια πετύχει επίδοση μεγαλύτερη από το κατώφλι πρόκρισης των 7,5 μ., σταματάει τις προσπάθειές του.

Να γράψετε πρόγραμμα σε γλώσσα προγραμματισμού Python το οποίο να υλοποιεί τα εξής:

4.1 Να διαβάζει το επώνυμο του μαθητή/τριας και να το αποθηκεύει στη λίστα ΟΝΟΜΑ των συμμετεχόντων μαθητών.

Μονάδες 5

4.2 Για κάθε μαθητή να διαβάζει την επίδοσή του ανά προσπάθεια και αν είναι μεγαλύτερη από το όριο πρόκρισης, να καταχωρίζει την επίδοση στη λίστα EPID.

Μονάδες 10

4.3 Να υπολογίζει και να εμφανίζει το πλήθος των μαθητών που προκρίθηκαν αλλά και το πλήθος των μαθητών που δεν προκρίθηκαν στον τελικό.

Μονάδες 5

4.4 Να εμφανίζει τα ονόματα όλων των μαθητών που συμμετείχαν.

Μονάδες 5

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
    print b
```

ΘΕΜΑ 4

```
1 # 4.1
2 ONOMA=[ ] # Αρχικοποιήσεις
3 EPID=[ ]
4 # 4.2
5 for i in range(30): # Βρόχος επανάληψης
6     n=raw_input("Δώσε όνομα μαθητή :")
7     ONOMA.append(n)
8     p1=input("Δώσε 1η προσπάθεια μαθητή :")
9     if p1 > 7.5 : # Δομή επιλογής
10         EPID.append(p1)
11     else :
12         p2=input("Δώσε 2η προσπάθεια μαθητή :")
13         if p2 > 7.5 :
14             EPID.append(p2)
15 # 4.3
16     print "Το πλήθος των μαθητών που προκρίθηκαν είναι :", len(EPID) # Εμφάνιση
17     αποτελεσμάτων
18     print "Το πλήθος των μαθητών που δεν προκρίθηκαν είναι :", 30 - len(EPID)
19 # 4.4
20     print "Οι μαθητές που συμμετείχαν είναι οι παρακάτω :"
21     for i in range(30):
22         print ONOMA[i]
```

ΘΕΜΑ 2

2.1 Να συμπληρωθούν τα τέσσερα (4) αριθμημένα κενά στο παρακάτω πρόγραμμα σε Python το οποίο δέχεται ακεραίους αριθμούς από τον χρήστη και τους προσθέτει σε μια λίστα, μόνο αν αυτοί δεν υπάρχουν ήδη. Το πρόγραμμα σταματά όταν εισαχθούν 15 ακέραιοι αριθμοί στη λίστα, και τότε τους εμφανίζει στην οθόνη.

```
count = __1__  
nums = []  
while __2__ < 15:  
    number = int(input("Δώστε έναν αριθμό "))  
    if number not in nums:  
        nums.append(number)  
        count = __3__ + 1  
print __4__
```

Μονάδες 8

2.2 Δίνονται τα παρακάτω τμήματα προγράμματος σε γλώσσα προγραμματισμού Python. Να γράψετε στο γραπτό σας το πλήθος των συμβόλων "+" που θα εμφανιστούν σε κάθε εκτέλεση.

α) for i in range(1,3): for j in range(1,3): print "+"	β) for i in range(1,4): for j in range(1,3): print "+"	γ) for i in range(1,6): for j in range(2,5): print "+"
--	--	--

Μονάδες 9

2.3 Να γράψετε στο γραπτό σας τα αποτελέσματα (True ή False) των παρακάτω λογικών εκφράσεων.

- α. $55 \neq 80$
- β. $63 \leq 23$
- γ. $(22 < 1) \text{ and } (32 > 11)$
- δ. $(13 < 10) \text{ or } (25 > 15)$

Μονάδες 8

ΘΕΜΑ 2

2.1

(1) 0

(2) count

(3) count

(4) nums

2.2

α) 4

β) 6

γ) 15

2.3

α. True

β. False

γ. False

δ. True

ΘΕΜΑ 4

Μια εταιρεία παροχής ενέργειας χρεώνει τους συνδρομητές της, ως προς την ποσότητα κατανάλωσης κιλοβατώραν KWh, μη κλιμακωτά, σύμφωνα με τον παρακάτω πίνακα:

Όγκος Κατανάλωσης σε KWh	Χρέωση ανά KWh
Έως 1600	0,2 ευρώ
Από 1601 μέχρι και 2000	0,3 ευρώ
Περισσότερα από 2000	0,4 ευρώ

Αν η χρέωση είναι μεγαλύτερη από εβδομήντα (200) ευρώ, τότε ο πελάτης δικαιούται έκπτωση 5%. Να γράψετε πρόγραμμα σε γλώσσα προγραμματισμού Python, το οποίο να υλοποιεί τα ακόλουθα:

4.1 Να διαβάζει από τον χρήστή τον όγκο των KWh που κατανάλωσε ο συνδρομητής. Δεν απαιτούνται έλεγχοι ορθότητας δεδομένων.

Μονάδες 2

4.2 Να καλεί συνάρτηση, η οποία να δέχεται ως παράμετρο τις KWh που κατανάλωσε ο συνδρομητής και να επιστρέφει τη χρέωση, χωρίς την έκπτωση, σύμφωνα με τον παραπάνω πίνακα. Να Γράψετε τη συνάρτηση που χρειάζεται.

Μονάδες 14

4.3 Να εμφανίζει τη χρέωση του συνδρομητή που προκύπτει από την παραπάνω συνάρτηση.

Μονάδες 2

4.4 Αν ο συνδρομητής δικαιούται έκπτωση, να εμφανίζει το ποσό της έκπτωσης, και την τελική χρέωση του πελάτη. Διαφορετικά, να εμφανίζει μήνυμα «Δεν δικαιούστε έκπτωση».

Μονάδες 7

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1 # 4.1
2 d = float(input("Δώστε κατανάλωση ενέργειας σε KWh: "))
3 # 4.2
4 def xreosh(a): # Επικεφαλίδα συνάρτησης
5     if a <= 1600 :
6         x = a*0.2
7     elif a <= 2000 :
8         x = a*0.3
9     else :
10        x = a*0.4
11    return x
12 p = xreosh(d) # Κλήση συνάρτησης
13 # 4.3
14 print "Η χρέωση σας είναι: ", p
15 # 4.4
16 if p > 200 :
17     ekp = p*0.05
18     print "Η έκπτωση είναι: ", ekp
19     print "Η τελική χρέωση είναι: ", p-ekp
20 else :
21     print "Δεν δικαιούστε έκπτωση."
```

ΘΕΜΑ 2

2.1 Δίνεται το παρακάτω τμήμα προγράμματος στη γλώσσα προγραμματισμού Python:

```
1 sum1=0
2 a=2
3 for i in range(10, 14):
4     if i%2==1:
5         a=a*3
6     else:
7         sum1=sum1+a
```

Επίσης δίνεται το παρακάτω υπόδειγμα-πίνακας τιμών:

	i	sum1	a
Αρχικές τιμές		...	...
1 ^η επανάληψη	...	...	...
...	...	...	...

Να συμπληρώσετε στο γραπτό σας τον παραπάνω πίνακα τιμών, προσθέτοντας μία νέα γραμμή στον πίνακα για κάθε επανάληψη του προγράμματος με τις νέες τιμές των μεταβλητών i, sum1 και a.

Μονάδες 15

2.2 Δίνονται τα παρακάτω τμήματα προγράμματος σε γλώσσα προγραμματισμού Python.

Να γράψετε το πλήθος των συμβόλων "@" που θα εμφανιστούν σε κάθε εκτέλεση.

α)	β)	γ)	δ)
for i in range(5,10,2): print "@"	for i in range(1,20): print "@"	for i in range(2,8): for j in range(2,8,2): print "@"	for i in range(3,6): for j in range(2,9): print "@"

Μονάδες 10

ΘΕΜΑ 2

2.1

	i	sum1	a
Αρχικές τιμές		0	2
1 ^η επανάληψη	10	2	
2 ^η επανάληψη	11		6
3 ^η επανάληψη	12	8	
4 ^η επανάληψη	13		18

2.2

α) 3

β) 19

γ) 18

δ) 21

ΘΕΜΑ 2ο

2.1. Να γράψετε τις παρακάτω μαθηματικές εκφράσεις σε γλώσσα προγραμματισμού Python:

1. $\frac{\sqrt{a}}{b^2+3}+c$

2. $\frac{x^2+c^2}{2}$

3. $x^2+\frac{x+a}{y^2}$

Μονάδες 15

2.2. Δίνεται το παρακάτω πρόγραμμα σε γλώσσα προγραμματισμού Python:

```
1  a=input("Δώσε τον 1ο ακέραιο αριθμό:")
2  b=input("Δώσε τον 2ο ακέραιο αριθμό:")
3  s=0
4  while b>0:
5      b,x=divmod(b,2)
6      if x==1:
7          s=s+a
8      a=a*2
9      print a,b,s
10 print s
```

Να γράψετε στο γραπτό σας τι εμφανίζετε στην οθόνη του υπολογιστή κατά την εκτέλεση του παραπάνω προγράμματος, δίνοντας τους αριθμούς 15 και 20 κατά την εκτέλεση των εντολών 1 και 2, αντίστοιχα.

Μονάδες 10

ΘΕΜΑ 2

2.1

1. $\text{math.sqrt}(a)/(b^{**2}+3)+c$

2. $(x^{**2}+c^{**2})/2$

3. $x^{**2}+(x+\alpha)/(y^{**2})$

2.2.

Δώσε τον 1ο ακέραιο αριθμό:15

Δώσε τον 2ο ακέραιο αριθμό:20

30 10 0

60 5 0

120 2 60

240 1 60

480 0 300

300

ΘΕΜΑ 2

2.1.

Να δώσετε την τιμή της παρακάτω έκφρασης, όπου $a=5$, $b=7$ και $c=3$, γράφοντας αναλυτικά τα βήματα που ακολουθείτε για τη λύση:

$\text{not}(b\%c+2==c) \text{ and } (a<5-c**2)$

Μονάδες 15

2.2. Δίνεται το παρακάτω πρόγραμμα σε γλώσσα προγραμματισμού Python:

```
1  x=[2,7,5,1,-2]
2  c=[3,-2,9,-2,25]
3  d=[0,0,0,0,0]
4  for i in range(0,4):
5      if i<=2:
6          d[i]=x[i]
7      else:
8          d[i]=c[i]
9  print d
```

Να γράψετε στο γραπτό σας τι εμφανίζεται κατά την εκτέλεση του παραπάνω προγράμματος .

Μονάδες 10

ΘΕΜΑ 2

2.1

$\text{not}(b\%c+2==c)$ and $(a<5-c**2) \rightarrow$

$\text{not}(7\%3+2==3)$ and $(5<5-3**2) \rightarrow$

$\text{not}(1+2==3)$ and $(5<5-9) \rightarrow$

$\text{not}(3==3)$ and $(5<-4) \rightarrow$

$\text{not}(\text{True})$ and $\text{False} \rightarrow$

False and $\text{False} \rightarrow \text{False}$

2.2.

[2, 7, 5, -2, 0]

ΘΕΜΑ 2

2.1.

Να αντιστοιχίσετε κάθε στοιχείο της στήλης Α του ακόλουθου πίνακα, με το κατάλληλο στοιχείο της στήλης Β. Τα στοιχεία της στήλης Β μπορείτε να τα χρησιμοποιήσετε καμία, μία ή περισσότερες από μία φορές.

Στήλη Α	Στήλη Β
1. and	Α. Ενσωματωμένη συνάρτηση
2. divmod	Β. Λογικός τελεστής
3. %	Γ. Αριθμητικός τελεστής
4. False	Δ. Λογική σταθερά (τιμή)
5. ==	Ε. Συγκριτικός τελεστής
	ΣΤ. Αριθμητική σταθερά (τιμή)

Μονάδες 10

2.2. Δίνεται το παρακάτω πρόγραμμα σε γλώσσα προγραμματισμού Python:

```
1 point=[]
2 x=152
3 y=x
4 while y!=0:
5     y,z=divmod(y,10)
6     point.append(z)
7     print z
8     print"-----"
9     s=0
10    for i in range(0,3):
11        s=s+point[i]
12    print "Άθροισμα=",s
```

Να γράψετε στο γραπτό σας τι εμφανίζεται κατά την εκτέλεση του παραπάνω προγράμματος .

Μονάδες 15

ΘΕΜΑ 2

2.1

1. Β
2. Α
3. Γ
4. Δ
5. Ε

2.2.

2
5
1

Άθροισμα= 8

ΘΕΜΑ 2

2.1.

Δίνεται το παρακάτω πρόγραμμα σε γλώσσα προγραμματισμού Python:

```
1  a=13
2  b=10
3  c,d=divmod(a,a+b)
4  z=a/b**2
5  h=(a/b)**2
6  f=a%b
7  print c,d,z,h,f
```

Να γράψετε στο γραπτό σας τι εμφανίζεται κατά την εκτέλεση του παραπάνω προγράμματος.

Μονάδες 10

2.2.

Δίνονται τα παρακάτω δύο προγράμματα σε γλώσσα προγραμματισμού Python. Στο 2^ο πρόγραμμα υπάρχουν τρία(3) κενά, τα οποία έχουν αριθμηθεί και υπογραμμιστεί. Να γράψετε στο γραπτό σας τον αριθμό κάθε κενού και δίπλα τι πρέπει να συμπληρωθεί, ώστε το 2^ο πρόγραμμα να εκτελεί την ίδια λειτουργία με το 1^ο πρόγραμμα:

1 ^ο πρόγραμμα	2 ^ο πρόγραμμα
<pre>1 a=input("Δώσε α:") 2 b=input("Δώσε β:") 3 if a>5: 4 if b==1: 5 print "*" 6 else: 7 print "***" 8 else: 9 if b==2: 10 print "***"</pre>	<pre>1 a=input("Δώσε α:") 2 b=input("Δώσε β:") 3 if ____ (1) ____: 4 print "*" 5 if ____ (2) ____: 6 print "***" 7 if ____ (3) ____: 8 print "***"</pre>

Μονάδες 15

ΘΕΜΑ 2

2.1

0 13 0 1 3

2.2.

1. $a > 5$ and $b == 1$
2. $a > 5$ and $b != 1$
3. $a \leq 5$ and $b == 2$

ΘΕΜΑ 4

Σε ένα διαγωνισμό του ΑΣΕΠ, για την πρόσληψη 50 υπαλλήλων σε θέσεις του Δημοσίου καταθέσανε αίτηση 450 πτυχιούχοι υποψήφιοι. Η πρόσληψη γίνεται βάση των μορίων που συμπληρώνει ο κάθε υποψήφιος. Τα μόρια που μπορεί να πάρει κάθε υποψήφιος είναι:

Βαθμός σε γραπτή εξέταση	(Β) Βαθμός<10= 0 μόρια 10<=Βαθμός<13= 2 μόρια 13<=Βαθμός<16= 4 μόρια 16<=Βαθμός<18= 6 μόρια 18<=Βαθμός<=20= 10 μόρια
Τίτλος σπουδών	(Π) Πτυχίο ΑΕΙ= 2 μόρια (Μ) Μεταπτυχιακό =4 μόρια (Δ) Διδακτορικό= 8 μόρια

Να αναπτυχθεί πρόγραμμα, σε γλώσσα προγραμματισμού Python, το οποίο:

4.1. Να διαβάζει από το πληκτρολόγιο για κάθε υποψηφίου το ονοματεπώνυμο του, το βαθμό που πήρε ο υποψήφιος στη γραπτή εξέταση που έδωσε στον ΑΣΕΠ και τον τελευταίο τίτλο σπουδών του. Να θεωρήσετε ότι τα δεδομένα είναι σωστά και δεν χρειάζεται οποιοσδήποτε έλεγχος.

Μονάδες 8

4.2. Να υπολογίζει τα συνολικά μόρια κάθε υποψηφίου, και την ένδειξη «ΕΠΙΤΥΧΩΝ», αν τα συνολικά μόρια του είναι μεγαλύτερα ή ίσα του 8.

Μονάδες 10

4.3 Να υπολογίζει και να εμφανίζει το σύνολο των επιτυχόντων και το σύνολο των επιλαχόντων, αν υπάρχουν. Επιλαχόντες είναι οι επιτυχόντες που είναι πάνω από τους 50.

Μονάδες 7

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1  sum_epitychon=0 #Αθροιστής επιτυχόντων
2  for i in range(0,450):
3      sum=0 #Αθροιστής μορίων κάθε υποψηφίου
4      o=raw_input("Δώσε όνομα υποψηφίου:")
5      grades=input("Δώσε τον βαθμό στις γραπτές εξετάσεις:")
6      print "Δώσε τελευταίο τίτλο σπουδών ",
7      study=raw_input("(Π)τυχίο ή (Μ)εταπτυχιακό ή (Δ)ιδακτορικό:")
8      if grades>=18 and grades<=20: #Έλεγχος μορίων βαθμού
9          sum=sum+10
10     elif grades>=16:
11         sum=sum+6
12     elif grades>=13:
13         sum=sum+4
14     elif grades>=10:
15         sum=sum+2
16     if study=='Π': #Έλεγχος για τον τελευταίο τίτλο σπουδών
17         sum=sum+2
18     elif study=='Μ':
19         sum=sum+4
20     else:
21         sum=sum+8
22     print "Συνολικά μόρια :",sum,
23     if sum>=8:
24         print "ΕΠΙΤΥΧΩΝ"
25         sum_epitychon=sum_epitychon+1
26 #Εμφάνιση συνόλου επιτυχόντων και επιλαχόντων
27 print "Σύνολο επιτυχόντων:",sum_epitychon
28 if sum_epitychon>50:
29     epilaxon=sum_epitychon-50
30     print "Σύνολο επιλαχόντων=",epilaxon
```


ΘΕΜΑ 4

Μία ποδοσφαιρική ομάδα έχει 18 παίκτες και την ενδιαφέρει ποιος παίκτης έχει πετύχει τα περισσότερα γκολ. Να αναπτυχθεί πρόγραμμα, σε γλώσσα προγραμματισμού Python, το οποίο:

4.1. Να διαβάζει από το πληκτρολόγιο τα ονόματα των παικτών και τα γκολ που έβαλαν σε μια αγωνιστική περίοδο. Να γίνεται έλεγχος δεδομένων ώστε ο αριθμός των γκολ που δίνεται από το πληκτρολόγιο να μην είναι αρνητικός. Στη περίπτωση που εισάγεται άκυρη τιμή, το πρόγραμμα θα πρέπει να ζητάει νέα.

Μονάδες 10

4.2. Να υπολογίζει τον παίκτη με τα περισσότερα γκολ, και να εμφανίζει το μήνυμα: «Ο όνομα παίκτη έβαλε τα περισσότερα γκολ, αριθμός γκολ »

Μονάδες 10

4.3. Να υπολογίζει και να εμφανίζει το σύνολο των γκολ που έχει πετύχει η ομάδα στην αγωνιστική περίοδο.

Μονάδες 5

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1  #Αρχική τιμή για τη max και την max_onoma
2  max=0
3  max_onoma=""
4  # Αρχική τιμή για τον αθροιστή των γκολ
5  sum=0
6  # Είσοδος δεδομένων και εύρεση μεγαλύτερου
7  for i in range(0,18):
8      # Δίνουμε το όνομα παίκτη και τα γκολ του
9      o=raw_input("Δώσε όνομα παίκτη: ")
10     gol=input("Δώσε τα συνολικά γκολ της αγωνιστικής περιόδου: ")
11     #Έλεγχος αν τα γκολ που πληκτρολογήσαμε είναι αρνητικός αριθμός
12     while gol<0:
13         print "Ο αριθμός πρέπει να μην είναι αρνητικός"
14         gol=input("Δώσε ξανά τα συνολικά γκολ της αγωνιστικής περιόδου: ")
15     sum=sum+gol
16     if gol>max:
17         max=gol
18         max_onoma=o
19 #Εκτύπωση αποτελεσμάτων
20 print "Ο παίκτης ",max_onoma," έβαλε τα περισσότερα γκολ ",max
21 print "Τα συνολικά γκολ της ομάδας είναι ",sum
```

ΘΕΜΑ 4

Το Υπουργείο Υγείας πραγματοποιεί έρευνα για την επιλογή ειδικότητας στους πτυχιούχους της Ιατρικής. Οι ειδικότητες που μπορούν να επιλέξουν οι πτυχιούχοι της Ιατρικής είναι: Π=παθολόγος ή Χ=χειρουργός. Να αναπτυχθεί πρόγραμμα, σε γλώσσα προγραμματισμού Python, το οποίο:

4.1. Να διαβάζει από το πληκτρολόγιο ονοματεπώνυμο πτυχιούχου και την ειδικότητα που θέλει να επιλέξει (Π ή Χ) και να τα καταχωρεί αντίστοιχα στις λίστες `onoma` και `protimisi`. Η καταχώρηση τελειώνει όταν πληκτρολογήσουμε * στο όνομα του πτυχιούχου. Να θεωρήσετε ότι τα δεδομένα είναι σωστά και δεν χρειάζεται έλεγχος εγκυρότητας των δεδομένων.

Μονάδες 7

4.2. Να υπολογίζει και να τυπώνει το σύνολο των προτιμήσεων ανά ειδικότητα.

Μονάδες 10

4.3. Να υπολογίζει και να εμφανίζει την ειδικότητα με το μεγαλύτερο αριθμό προτιμήσεων και στην περίπτωση ισοβαθμίας να εμφανίζεται το μήνυμα "Οι Παθολόγοι και οι Χειρουργοί έχουν την ίδια προτίμηση".

Μονάδες 8

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1  # Αρχικοποίηση λιστών
2  onoma=[]
3  protimisi=[]
4
5  # Είσοδος δεδομένων από το πληκτρολόγιο μέχρι να πληκτρολογήσουμε * στο όνομα
6  o=raw_input("Δώσε όνομα πτυχιούχου Ιατρικής. Με * τερματίζει η καταχώρηση: ")
7  while o!="*":    # Έλεγχος αν στο όνομα πληκτρολογήσαμε *
8      onoma.append(o)
9      e=raw_input("Δώσε ειδικότητα Π=Παθολόγος Χ=Χειρουργός :")
10     protimisi.append(e)
11     o=raw_input("Δώσε όνομα πτυχιούχου Ιατρικής. Με * τερματίζει η καταχώρηση: ")
12 #Μηδενισμός μετρητών Παθολόγων και Χειρουργών
13 m_path=0
14 m_xeir=0
15 n=len(protimisi)
16 #Έλεγχος κάθε στοιχείου της λίστας protimisi με την ειδικότητα που καταχωρήθηκε
17 for i in range(0,n):
18     if protimisi[i]=="Π":    # Έλεγχος για την επιλογή Παθολόγος
19         m_path=m_path+1      # Αύξηση του μετρητή m_path κατά 1
20     else:                    # Διαφορετικά είναι Χειρουργός
21         m_xeir=m_xeir+1     # Αύξηση του μετρητή m_xeir κατά 1
22
23 # Εμφάνιση των αποτελεσμάτων
24 print "ΣΥΝΟΛΟ ΠΡΟΤΙΜΗΣΕΩΝ ΑΝΑ ΕΙΔΙΚΟΤΗΤΑ"
25 print "Παθολόγοι:",m_path
26 print "Χειρουργοί:",m_xeir
27
28 # Εμφάνιση της Ειδικότητας με τις περισσότερες προτιμήσεις
29 if m_path>m_xeir:
30     print "Οι Παθολόγοι έχουν τις περισσότερες προτιμήσεις"
31 elif m_xeir>m_path:
32     print "Οι Χειρουργοί έχουν τις περισσότερες προτιμήσεις"
33 else:
34     print "Οι Παθολόγοι και οι Χειρουργοί έχουν την ίδια προτίμηση"
```


ΘΕΜΑ 4

Ένα σούπερ μάρκετ έχει 450 προϊόντα. Για την αποτελεσματική διαχείριση των προϊόντων το σούπερ μάρκετ καταγράφει το όνομα και το πλήθος των τεμαχίων του κάθε προϊόντος.

Να αναπτυχθεί πρόγραμμα, σε γλώσσα προγραμματισμού Python, το οποίο:

4.1. Να καταχωρεί το όνομα και το πλήθος των διαθέσιμων τεμαχίων κάθε προϊόντος στις λίστες `on_pr` και `pl` αντίστοιχα.

Μονάδες 5

4.2. Να εμφανίζει τον τίτλο «ΠΡΟΪΟΝΤΑ ΓΙΑ ΠΑΡΑΓΓΕΛΙΑ» και στη συνέχεια τα ονόματα των προϊόντων που πρέπει να παραγγελθούν, όταν το πλήθος των διαθέσιμων τεμαχίων κάποιου προϊόντος είναι λιγότερο από 10.

Μονάδες 10

4.3. Να υπολογίζει και να εμφανίζει το πλήθος όλων των προϊόντων που είναι για παραγγελία.

Μονάδες 10

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1  #Αρχικοποίηση λιστών
2  on_pr=[]
3  pl=[]
4  #Καταχώρηση δεδομένων στις αντίστοιχες λίστες
5  for i in range(0,450):
6      p=raw_input("Δώσε όνομα προϊόντος: ")
7      on_pr.append(p)
8      te=input("Δώσε τεμάχια προϊόντος: ")
9      pl.append(te)
10     print "_____ "
11 n=0 #Αρχικοποίηση του μετρητή προϊόντων για παραγγελία
12 print "ΠΡΟΪΟΝΤΑ ΓΙΑ ΠΑΡΑΓΓΕΛΙΑ"
13 for i in range(0,450):
14     if pl[i]<10: #Έλεγχος προϊόντων για παραγγελία
15         print on_pr[i]#Εμφάνιση ονόματος προϊόντος για παραγγελία
16         n=n+1 #Μετρητής προϊόντων για παραγγελία
17     print"_____ "
18 #Εμφάνιση του πλήθους των προϊόντων για παραγγελία
19 print "Σύνολο προϊόντων για παραγγελία=", n
```

ΘΕΜΑ 4

Σε ένα διαγωνισμό του Δημοσίου συμμετείχαν 3.000 διαγωνιζόμενοι, οι οποίοι εξετάστηκαν στη «Έκθεση». Επιτυχών είναι αυτός που είχε βαθμό μεγαλύτερο του 50. Να αναπτυχθεί πρόγραμμα, σε γλώσσα προγραμματισμού Python, το οποίο:

4.1. Να διαβάζει το ονοματεπώνυμο και τον βαθμό κάθε υποψηφίου και να καταχωρεί τα δεδομένα αντίστοιχα στις λίστες `ονομα[3000]` και `b[3000]`. Αποδεκτός βαθμός είναι από 0 μέχρι 100. Στην περίπτωση που εισάγεται μη αποδεκτό βαθμό, το πρόγραμμα πρέπει να ζητάει άλλον μέχρι να δοθεί τιμή εντός των ορίων.

Μονάδες 10

4.2. Να υπολογίζει και να εμφανίζει τα ονόματα των υποψηφίων και το μήνυμα «ΕΠΙΤΥΧΩΝ» ή «ΑΠΟΤΥΧΩΝ».

Μονάδες 10

4.3. Να υπολογίζει και να εμφανίζει το σύνολο των επιτυχόντων.

Μονάδες 5

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```


ΘΕΜΑ 4

```
1  # Αρχικοποίηση λιστών
2  onoma=[]
3  b=[]
4  # Είσοδος δεδομένων κάθε υποψηφίου
5  for i in range(3000):
6      o=raw_input("Δώσε όνομα διαγωνιζόμενου: ")
7      onoma.append(o)
8      ba=input("Δώσε το βαθμό του στην Έκθεση: ")
9      while ba<=0 or ba>100:
10         print "Έδωσες λάθος βαθμό..."
11         ba=input("Ξαναδώσε το βαθμό του στην Έκθεση: ")
12     b.append(ba)
13 print "_____ "
14 # Εκτύπωση όλων των υποψηφίων με το μήνυμα ΕΠΙΤΥΧΩΝ ή ΑΠΟΤΥΧΩΝ
15 m=0  #Μετρητής επιτυχόντων
16 for i in range(3000):
17     print onoma[i],b[i]
18     if b[i]>50:
19         print "ΕΠΙΤΥΧΩΝ"
20         m=m+1  #Αύξηση του μετρητή των επιτυχόντων κατά μία μονάδα
21     else:
22         print "ΑΠΟΤΥΧΩΝ"
23 #Εκτύπωση του συνόλου των επιτυχόντων
24 print "Σύνολο επιτυχόντων:",m
```

ΘΕΜΑ 4

Μια εταιρεία ύδρευσης χρεώνει τους συνδρομητές της, ως προς την ποσότητα κατανάλωσης κυβικών (μ^3) νερού για οικιακή χρήση, κλιμακωτά, σύμφωνα με τον ακόλουθο πίνακα:

Όγκος Κατανάλωσης σε κυβικά	Χρέωση ανά κυβικό
Έως και 20	0,3 ευρώ
Πάνω από 20 μέχρι και 35	1,8 ευρώ
Περισσότερα από 35	3,2 ευρώ

Αν η χρέωση είναι μεγαλύτερη από πενήντα (50) ευρώ, τότε ο συνδρομητής δικαιούται έκπτωση 7%. Να γράψετε πρόγραμμα σε γλώσσα προγραμματισμού Python, το οποίο να υλοποιεί τα ακόλουθα:

4.1 Να διαβάζει τον όγκο των μ^3 νερού που κατανάλωσε ο συνδρομητής. Να γίνεται έλεγχος ορθότητας δεδομένων έτσι ώστε η κατανάλωση να είναι μεγαλύτερη ή ίση του μηδενός.

Μονάδες 5

4.2 Να υπολογίζει και να εμφανίζει τη χρέωση, χωρίς την έκπτωση, σύμφωνα με τον παραπάνω πίνακα (κλιμακωτά).

Μονάδες 13

4.3 Αν ο πελάτης δικαιούται έκπτωση, να εμφανίζει το ποσό της έκπτωσης, και την τελική χρέωση του πελάτη. Διαφορετικά, να εμφανίζει μήνυμα στην οθόνη «Δεν δικαιούστε έκπτωση».

Μονάδες 7

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1 # 4.1
2 d = float(input("Δώστε κατανάλωση νερού σε κυβικά μέτρα: "))
3 # έλεγχος ορθότητας δεδομένων
4 while d < 0 :
5     d = float(input("Δώστε κατανάλωση νερού σε κυβικά μέτρα: "))
6 # 4.2
7 if d <= 20 :
8     x = d*0.3
9 elif d <= 35 :
10    x = 20*0.3 + (d-20)*1.8 # κλιμακωτή χρέωση
11 else :
12    x = 20*0.3 + 15*1.8 + (d-35)*3.2
13 print "Η χρέωση σας είναι: ", x
14 # 4.3
15 if x > 50 : # υπολογισμός έκπτωσης
16     ekp = x*0.07
17     print "Η έκπτωση είναι: ", ekp
18     print "Η τελική χρέωση είναι: ", x-ekp
19 else :
20     print "Δεν δικαιούστε έκπτωση."
```

ΘΕΜΑ 2

2.1 Να συμπληρωθούν τα αριθμημένα κενά (1-8) στον παρακάτω κώδικα σε γλώσσα προγραμματισμού Python, έτσι ώστε το πρόγραμμα να διαβάζει ακέραιους αριθμούς και να τους εισάγει σε δύο λίστες A και B, των 5 θέσεων η καθεμία. Στη συνέχεια να δημιουργεί και να τυπώνει στην οθόνη μια λίστα C, 5 θέσεων, τα στοιχεία της οποίας προκύπτουν από το γινόμενο των αντίστοιχων στοιχείων από τις λίστες A και B.

```
1  A = []
2  B = []
3  C = []
4  for i in range(5):
5      x = int(input("Δώσε αριθμό για τη λίστα A: "))
6      A.append(_(1)_)
7  for i in range(_(2)_):
8      y = _(3)
9      _(4)
10  _(5) range(_(6)_):
11      _(7)
12  print(_(8)_)
```

Μονάδες 16

2.2 Να γράψετε στο γραπτό σας τις εντολές σε Python για να εκχωρήσετε σε δύο μεταβλητές a, b τις τιμές 1 και 0, αντίστοιχα και έπειτα να αντιμεταθέσετε τις τιμές των μεταβλητών αυτών.

Μονάδες 9

ΘΕΜΑ 2

2.1

1 – x

2 – 5

3 – int(input("Δώσε αριθμό για λίστα B: "))

4 – B.append(y)

5 – for i in

6 – 5

7 – C.append(A[i] * B[i])

8 – C

2.2

a, b = 1, 0

a, b = b, a

ΘΕΜΑ 4

Σε έναν αγώνα άλματος εις μήκος συμμετέχουν στον προκριματικό 24 αθλήτριες. Στον τελικό αγώνα θα συμμετάσχουν μόνο οι αθλήτριες οι οποίες θα πετύχουν επίδοση μεγαλύτερη ή ίση από 4,5 μέτρα που είναι το όριο πρόκρισης. Κάθε αθλήτρια έχει μέχρι τρεις (3) προσπάθειες για να πετύχει το όριο πρόκρισης. Αν πετύχει σε μία προσπάθεια, δεν συνεχίζει τις υπόλοιπες προσπάθειες:

Να γράψετε πρόγραμμα σε γλώσσα προγραμματισμού Python, το οποίο:

4.1 Για κάθε αθλήτρια να διαβάζει το επώνυμο της και διαδοχικά την επίδοση των προσπαθειών της μέχρι να πετύχει την κατάλληλη επίδοση ή να συμπληρωθεί ο αριθμός των τριών προσπαθειών που δικαιούται.

Μονάδες 14

4.2 Για κάθε αθλήτρια να εμφανίζει στην οθόνη το επώνυμο της και αν προκρίθηκε την επίδοση πρόκρισής της, διαφορετικά να εμφανίζει το μήνυμα “ΜΗ ΠΡΟΚΡΙΣΗ”.

Μονάδες 6

4.3 Να υπολογίζει και να εμφανίζει στην οθόνη το πλήθος των προκριμένων αθλητριών.

Μονάδες 5

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
    print b
```

ΘΕΜΑ 4

```
1 plithos = 0 # αρχικοποίηση μετρητή
2 #Γ1
3 for i in range(24): # εξωτερικός βρόχος επανάληψης
4     onoma= raw_input("Δώστε επώνυμο ")
5     prosp= 1
6     epidosi= input("Δώστε επίδοση ")
7     while prosp < 3 and epidosi < 4.5:
8         epidosi= input("Δώστε νέα επίδοση ")
9         prosp += 1
10 #Γ2
11     print onoma
12     if epidosi>=4.5: # έλεγχος προκριθέντων αθλητριών
13         print "Πρόκριση με: ", epidosi
14         #Γ3
15         plithos+=1
16     else:
17         print "ΜΗ ΠΡΟΚΡΙΣΗ"
18 #Γ3
19 print "Το πλήθος των αθλητριών που προκρίθηκαν είναι: ", plithos
```

ΘΕΜΑ 2

2.1 Να γράψετε τις τιμές που εισάγονται στη λίστα σε κάθε περίπτωση σαν αποτέλεσμα της εκτέλεσης των ακόλουθων τμημάτων κώδικα στη γλώσσα προγραμματισμού Python:

α) lista = [] for i in range (2,7,1): lista.append(i**2)	β) lista = [] for i in range (4,12,2): lista.append(i**2)
---	--

Μονάδες 10

2.2 Δίνεται το παρακάτω τμήμα προγράμματος σε Python:

```
for i in range (A, B, Γ):
```

```
    print i
```

Για καθεμιά από τις παρακάτω περιπτώσεις, να γράψετε τις τιμές των A, B, Γ, έτσι ώστε το αντίστοιχο τμήμα προγράμματος να εμφανίζει όλους:

α. τους ακέραιους από 1 μέχρι και 60 (αύξουσα διάταξη)

β. τους ακέραιους από 70 μέχρι και 30 (φθίνουσα διάταξη)

γ. τους περιττούς ακέραιους από 51 μέχρι και 131 (αύξουσα διάταξη)

δ. τους ακέραιους από -30 μέχρι και -10 (αύξουσα διάταξη)

ε. τους θετικούς ακέραιους που είναι μικρότεροι του 300 και πολλαπλάσιοι του 6 (αύξουσα διάταξη).

Μονάδες 15

ΘΕΜΑ 2

2.1

α) 4 9 16 25 36

β) 16 36 64 100

2.2

α . $A=1$, $B=61$, $\Gamma=1$

β . $A=70$, $B=29$, $\Gamma=-1$

γ . $A=51$, $B=132$, $\Gamma=2$

δ . $A=-30$, $B=-9$, $\Gamma=1$

ϵ . $A=6$, $B=300$, $\Gamma=6$

ΘΕΜΑ 4

Σε έναν αθλητικό-φοιτητικό αγώνα σφαιροβολίας συμμετέχουν στον προκριματικό γύρο 32 φοιτητές. Στον τελικό αγώνα θα προκριθούν μόνο οι φοιτητές οι οποίοι θα πετύχουν επίδοση μεγαλύτερη ή ίση από 15 μέτρα που είναι το όριο πρόκρισης. Κάθε φοιτητής έχει μέχρι πέντε (5) προσπάθειες για να πετύχει το όριο πρόκρισης. Αν πετύχει σε μία προσπάθεια, δεν συνεχίζει τις υπόλοιπες προσπάθειες:

Να γράψετε πρόγραμμα σε γλώσσα προγραμματισμού Python, το οποίο:

4.1 Για κάθε φοιτητή να διαβάζει το επώνυμο του και διαδοχικά την επίδοση των προσπαθειών του μέχρι να πετύχει την κατάλληλη επίδοση ή να συμπληρωθεί ο αριθμός των πέντε προσπαθειών που δικαιούται.

Μονάδες 12

4.2 Για κάθε φοιτητή να εμφανίζει το επώνυμο του και αν προκρίθηκε την επίδοση πρόκρισής του, διαφορετικά να εμφανίζει το μήνυμα “ΔΕΝ ΠΡΟΚΡΙΝΕΣΑΙ”.

Μονάδες 6

4.3 Να υπολογίζει και να εμφανίζει το πλήθος των προκριμένων φοιτητών καθώς και το μέσο όρο των επιδόσεων πρόκρισης που πέτυχαν. Θεωρείστε ότι υπάρχει τουλάχιστον ένας φοιτητής ο οποίος προκρίθηκε στον τελικό αγώνα.

Μονάδες 7

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
    print b
```

ΘΕΜΑ 4

```
1  plithos = 0 # αρχικοποίηση μετρητή
2  sumf=0.0
3  #Γ1
4  for i in range(3): # εξωτερικός βρόχος επανάληψης
5      onoma= raw_input("Δώστε επώνυμο ")
6      prosp= 1
7      epidosi= input("Δώστε επίδοση ")
8      while prosp < 5 and epidosi < 15:
9          epidosi= input("Δώστε νέα επίδοση ")
10         prosp += 1
11     #Γ2
12     print onoma
13     if epidosi>=15: # έλεγχος προκριθέντων φοιτητών
14         print epidosi
15         #Γ3
16         plithos+=1
17         sumf=sumf+epidosi
18     else:
19         print "ΔΕΝ ΠΡΟΚΡΙΝΕΣΑΙ"
20     #Γ3
21     print plithos
22     print sumf/plithos
```

ΘΕΜΑ 2

2.1 Δίνεται το παρακάτω τμήμα προγράμματος σε Python:

```
1  x = int(input("Δώστε έναν ακέραιο "))
2  y = int(input("Δώστε έναν ακέραιο "))
3  if y < x :
4      z = x
5  else :
6      z = y
7  while z != 0 :
8      z = x % y
9      x = y
10     y = z
```

Επίσης, δίνεται το παρακάτω υπόδειγμα-πίνακας τιμών. Να συμπληρώσετε τον πίνακα με όσες τιμές απαιτούνται για τις μεταβλητές x, y και z εκτελώντας το τμήμα προγράμματος για αρχικές τιμές εισόδου x = 95 και y = 30.

x	y	z
95	30	...
...	...	...

Μονάδες 14

2.2 Δίνεται το παρακάτω τμήμα προγράμματος σε Python:

```
1  for i in range(2,11,4) :
2      x = int(input("Πληκτρολογήστε έναν αριθμό "))
3      if x == 0 :
4          print "Ίσος με μηδέν"
5      if x > 0 :
6          print "Θετικός"
7      if x < 0 :
8          print "Αρνητικός"
```

α. Να γράψετε πόσες φορές θα εκτελεστεί η δομή επανάληψης for.

β. Να γράψετε το τμήμα προγράμματος με χρήση της δομής επανάληψης while αντί της δομής επανάληψης for.

Μονάδες 11

ΘΕΜΑ 2

2.1

x	y	z
95	30	30
30	5	5
5	0	0

2.2

α. 3 φορές

β.

```
1 i=2
2 while i<11 :
3     x = int(input("Πληκτρολογήστε έναν αριθμό "))
4     if x == 0 :
5         print "Ίσος με μηδέν"
6     if x > 0 :
7         print "Θετικός"
8     if x < 0 :
9         print "Αρνητικός"
10    i = i + 4
```

ΘΕΜΑ 4

Σε έναν αγώνα Formula-2 συμμετέχουν στις κατατακτήριες δοκιμές 20 οδηγοί. Στον τελικό αγώνα θα προκριθούν μόνο οι οδηγοί οι οποίοι θα πετύχουν επίδοση σε έναν γύρο στην πίστα οδήγησης μικρότερη ή ίση από 2,5 δευτερόλεπτα που είναι το όριο πρόκρισης. Κάθε οδηγός έχει δέκα (10) προσπάθειες για να πετύχει το όριο πρόκρισης. Αν πετύχει σε μία προσπάθεια, δεν συνεχίζει με τις υπόλοιπες προσπάθειες.

Να αναπτύξετε πρόγραμμα σε γλώσσα προγραμματισμού Python, το οποίο:

4.1 Για κάθε οδηγό να διαβάζει το επώνυμο του και διαδοχικά την επίδοση των προσπαθειών του μέχρι να συμπληρωθεί ο αριθμός των δέκα προσπαθειών που δικαιούται.

Μονάδες 10

4.2 Για κάθε οδηγό να εμφανίζει το επώνυμο του και αν προκρίθηκε την επίδοση πρόκρισής του, διαφορετικά να εμφανίζει το μήνυμα “ΕΚΤΟΣ ΤΕΛΙΚΟΥ”.

Μονάδες 5

4.3 Να υπολογίζει και να εμφανίζει το πλήθος των προκριμένων οδηγών.

Μονάδες 3

4.4 Να βρίσκει και να εμφανίζει το όνομα του οδηγού με τον μικρότερο χρόνο πρόκρισης και τον χρόνο αυτό. Να θεωρήσετε ότι αυτός ο χρόνος είναι μοναδικός.

Μονάδες 7

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
    print b
```

ΘΕΜΑ 4

```
1  plithos = 0 # αρχικοποίηση μετρητή
2  min_epid = 100
3  #Γ1
4  for i in range(20): # εξωτερικός βρόχος επανάληψης
5      onoma= raw_input("Δώστε επώνυμο ")
6      prosp= 1
7      epidosi= input("Δώστε επίδοση σε sec ")
8      while prosp < 10 and epidosi > 2.5:
9          epidosi= input("Δώστε νέα επίδοση σε sec ")
10         prosp += 1
11     #Γ2
12     print onoma
13     if epidosi<=2.5: # έλεγχος προκριθέντων οδηγών
14         print epidosi
15         #Γ3
16         plithos+=1
17     else:
18         print "ΔΕΝ ΠΡΟΚΡΙΝΕΣΑΙ"
19     #Γ3
20     print plithos
21     #Γ4
22     if epidosi<= min_epid:
23         min_epid = epidosi
24         min_onoma = onoma
25     print min_onoma
26     print min_epid
```

ΘΕΜΑ 2

2.1 Δίνεται το παρακάτω τμήμα προγράμματος σε Python:

```
1 x = int(input ("Δώστε έναν αριθμό" ) )
2 if x == 0 :
3     print "Ίσος με το μηδέν"
4 if x < 0 :
5     print "Αρνητικός"
6 if x > 0 :
7     print "Θετικός"
```

α. Να γράψετε τι θα εμφανίσει στην οθόνη αν δοθεί για τιμή εισόδου $x=-8$.

β. Να γράψετε το παραπάνω τμήμα προγράμματος με χρήση της δομής πολλαπλής επιλογής if ... elif αντί των απλών δομών επιλογής if.

Μονάδες 11

2.2 Δίνεται το παρακάτω τμήμα προγράμματος σε γλώσσα προγραμματισμού Python, το οποίο υπολογίζει και εμφανίζει το άθροισμα των ακεραίων αριθμών που είναι πολλαπλάσιοι του 4 και βρίσκονται στο διάστημα από το 1 έως το 80.

```
1 x=1
2 s= (1)
3 while x (2) 80:
4     if x (3) 4== (4) :
5         s= (5) +x
6     x=x+ (6)
7 print "Το άθροισμα ισούται με: ", (7)
```

Στο πρόγραμμα υπάρχουν επτά υπογραμμισμένα αριθμημένα κενά. Να γράψετε τους αριθμούς 1, 2, 3, 4, 5, 6 και 7 που αντιστοιχούν στα κενά του παραπάνω προγράμματος και δίπλα σε κάθε αριθμό ότι πρέπει να συμπληρωθεί για να υλοποιείται ορθά το τμήμα προγράμματος.

Μονάδες 14

ΘΕΜΑ 2

2.1

α. Αρνητικός

β.

```
1 x = int(input("Δώστε έναν αριθμό "))
2 if x == 0 :
3     print "Ισος με το μηδέν"
4 elif x < 0 :
5     print "Αρνητικός"
6 else :
7     print "Θετικός"
```

2.2

1. 0

2. <=

3. %

4. 0

5. s

6. 1

7. s

ΘΕΜΑ 4

Σε μία κινηματογραφική ταινία σε σινεμά το εισιτήριο εισόδου είναι για τους ενήλικες επτά (7) ευρώ και για τα παιδιά τέσσερα (4) ευρώ. Η χωρητικότητα του σινεμά είναι διακόσια (200) άτομα. Να αναπτύξετε πρόγραμμα σε γλώσσα προγραμματισμού Python, το οποίο να υλοποιεί τα παρακάτω:

4.1 Για κάθε άτομο να διαβάζει την τιμή “Ε” για ενήλικας ή “Ρ” για παιδί. Η διαδικασία της εισόδου θεατών να τερματίζεται, όταν εισαχθεί η τιμή “TELOS”. Δεν απαιτείται έλεγχος εγκυρότητας εισαγωγής δεδομένων.

Μονάδες 8

4.2 Να υπολογίζει και να εμφανίζει στην οθόνη τα συνολικά έσοδα του κινηματογράφου και το πλήθος των κενών θέσεων.

Μονάδες 9

4.3 Τέλος, να υπολογίζει και να εμφανίζει το ποσοστό των παιδιών στο σύνολο των θεατών. Θεωρήστε ότι υπάρχει τουλάχιστον ένας θεατής-παιδί.

Μονάδες 8

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1  capacity = 200 # αρχικοποιήσεις
2  total = 0
3  total_pe = 0
4  total_pp = 0
5  th=raw_input("Δώστε Ε για Ενήλικας ή Ρ για Παιδί ή ΤΕΛΟΣ για τερματισμό: ")
6
7  while th != "ΤΕΛΟΣ" : # 4.1
8      if th == "Ε":
9          total_pe = total_pe + 1
10         total = total + 7
11     else:
12         total_pp = total_pp + 1
13         total = total + 4
14
15     capacity = capacity - 1 # 4.2
16
17     th=raw_input("Δώστε Ε για Ενήλικας ή Ρ για Παιδί ή ΤΕΛΟΣ για τερματισμό: ")
18
19     pososto = float(total_pp)/(total_pe+total_pp) * 100 # 4.3
20     print "Συνολικά έσοδα κινηματογράφου: ",total # έξοδος αποτελεσμάτων
21     print "Κενές θέσεις: ", capacity
22     print "Ποσοστό παιδιών: ",pososto, "%"
```

ΘΕΜΑ 2

2.1 α) Να γράψετε το παρακάτω τμήμα προγράμματος, χρησιμοποιώντας την εντολή επανάληψης for αντί της εντολής επανάληψης while έτσι ώστε να εμφανίζει στην οθόνη τα ίδια αποτελέσματα.

```
1 i=3
2 while i <=15:
3     print i,
4     i=i+2
```

β) Να γράψετε και τα αποτελέσματα που εμφανίζονται στην οθόνη.

Μονάδες 11

2.2 Δίνεται το παρακάτω τμήμα προγράμματος σε γλώσσα προγραμματισμού Python, το οποίο υπολογίζει και εμφανίζει το γινόμενο των ακераίων αριθμών που είναι πολλαπλάσιοι του 3 και βρίσκονται στο διάστημα από το 1 έως το 30.

```
1 x=1
2 p= (1)
3 while x (2) 100:
4     if x (3) 3== (4) :
5         p= (5) *x
6         x=x+ (6)
7     print "Το γινόμενο ισούται με: ", (7)
```

Στο πρόγραμμα υπάρχουν επτά υπογραμμισμένα αριθμημένα κενά. Να γράψετε τους αριθμούς 1, 2, 3, 4, 5, 6 και 7 που αντιστοιχούν στα κενά του παραπάνω προγράμματος και δίπλα σε κάθε αριθμό ότι πρέπει να συμπληρωθεί για να υλοποιείται ορθά το τμήμα προγράμματος.

Μονάδες 14

ΘΕΜΑ 2

2.1

α)

```
1  for i in range (3, 16, 2) :  
2      print i,
```

β. 3 5 7 9 11 13 15

2.2

1. **1**

2. **<=**

3. **%**

4. **0**

5. **p**

6. **1**

7. **p**

ΘΕΜΑ 4

Η σχεδίαση για τον παιδικό εμβολιασμό ενός Κέντρου Υγείας σε έναν Δήμο για το έτος 2022 δίνεται από τον παρακάτω πίνακα:

Τύπος Εμβολίου	Ηλικία
1. Ηπατίτιδα Β	Από 1 έως και 3
2. Τέτανος	Από 4 έως και 7
3. Πνευμονιόκοκκος	Από 8 έως και 10
4. Μηνιγγίτιδα	Άνω των 10

Να αναπτύξετε πρόγραμμα σε γλώσσα προγραμματισμού Python το οποίο να υλοποιεί τα παρακάτω:

4.1 Να διαβάζει για κάθε παιδί που πρόκειται να εμβολιαστεί: Την ηλικία (ακέραια τιμή), το φύλο ("Α" για αγόρι και "Κ" για κορίτσι) και τον ΑΜΚΑ (Αριθμός Μητρώου Κοινωνικής Ασφάλισης) ως συμβολοσειρά. Δεν απαιτείται έλεγχος εγκυρότητας δεδομένων.

Η εισαγωγή των δεδομένων να τερματίζεται, όταν δοθεί ηλικία μεγαλύτερη του 17. Θεωρήστε ότι θα προσέλθει τουλάχιστον ένα παιδί για εμβολιασμό.

Μονάδες 8

4.2 Για κάθε παιδί, να βρίσκει και να εμφανίζει στην οθόνη τον τύπο του εμβολίου σύμφωνα με τον παραπάνω πίνακα. Επίσης, να εμφανίζει τον ΑΜΚΑ του παιδιού.

Μονάδες 9

4.3 Να υπολογίζει και να εμφανίζει στην οθόνη το ποσοστό των κοριτσιών στο σύνολο των παιδιών που θα εμβολιαστούν.

Μονάδες 8

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
    print b
```

ΘΕΜΑ 4

```
1  pl=0 # πλήθος όλων
2  plg=0 # πλήθος κοριτσιών
3  ilikia=int(input('Δώστε την ηλικία ')) # 4.1
4  while ilikia<=17: # βρόχος επανάληψης
5      pl=pl+1
6      fylo=raw_input('Δώστε το φύλο ')
7      while fylo!='A' and fylo!='K':
8          fylo=raw_input('Δώστε το φύλο ')
9      amka=raw_input('Δώστε τον ΑΜΚΑ ')
10     # 4.2
11     if ilikia<=3: # η ηλικία είναι μικρότερη ή ίση του 17
12         typos=1
13     elif ilikia<=7:
14         typos=2
15     elif ilikia<=10:
16         t=3
17     else:
18         typos=4
19
20     print 'Ο ασφαλισμένος με ΑΜΚΑ',amka,'θα κάνει εμβόλιο τύπου',typos
21     if fylo=='K':
22         plg=plg+1
23     ilikia=int(input('Δώστε την ηλικία '))
24     # 4.3
25     pososto=plg/float(pl)*100
26     print 'Το ποσοστό των κοριτσιών που εμβολιάστηκαν είναι',pososto,'%'
```

ΘΕΜΑ 2

2.1. Να γραφεί ως εντολή της Python η ακόλουθη αριθμητική έκφραση. Τι τύπου είναι η μεταβλητή z , αν οι μεταβλητές b, x, y είναι ακέραιες;

$$z = y \cdot \frac{\sqrt{b}}{1+x^2}$$

Μονάδες 10

2.2 Δίνεται το ακόλουθο πρόγραμμα σε Python:

```
1  sum=0
2  for i in range(5, -1, -2):
3      x=0
4      while x >= 0 and x <= 4:
5          sum= sum + x
6          x= x+2
7  print sum
```

A. Πόσες φορές θα εκτελεστεί η επανάληψη for στη γραμμή 2;

B. Ποιες τιμές θα πάρει διαδοχικά η μεταβλητή i ;

Μονάδες 15

ΘΕΜΑ 2

2.1

$$z = y * \text{math.sqrt}(b)/(1 + x ** 2)$$

Η μεταβλητή z είναι αριθμός κινητής υποδιαστολής (float)

2.2

A. Η επανάληψη for στη γραμμή 2 θα εκτελεστεί 3 φορές.

B. Το i θα πάρει διαδοχικά τις τιμές 5, 3, 1.

ΘΕΜΑ 4

Ένα ζευγάρι που πρόκειται να προχωρήσει σε αγορά κατοικίας, εξετάζει ποιο από τα επτά (7) διαμερίσματα που έχει δει είναι η πιο συμφέρουσα επιλογή. Η βαθμολογία κάθε διαμερίσματος υπολογίζεται αθροιστικά βάσει των κριτηρίων του ακόλουθου πίνακα:

Επιφάνεια διαμερίσματος (σε τ.μ.)	Βαθμός	Κόστος ανά τ.μ. (σε ευρώ)	Βαθμός	Όροφος	Βαθμός
≥ 120	2	≤ 6	3	1 – 4	1
< 120	0	> 6	0	0 (Ισόγειο) ή ≥ 5	0

Για παράδειγμα, ένα διαμέρισμα 130 τ.μ., στον 2^ο όροφο που κοστίζει 7 ευρώ ανά τ.μ., θα βαθμολογηθεί με 3 (2 βαθμοί για την επιφάνεια, 0 βαθμοί για το κόστος ανά τ.μ., συν 1 βαθμό για τον όροφο).

Να αναπτυχθεί πρόγραμμα σε Python το οποίο να:

4.1 Διαβάζει την περιοχή στην οποία βρίσκεται κάθε διαμέρισμα, το κόστος αγοράς του ανά τ.μ. σε ευρώ, την επιφάνειά του σε τ.μ και τον όροφο στον οποίο βρίσκεται (0 για ισόγειο). Θεωρείστε ότι όλα τα διαμερίσματα βρίσκονται σε διαφορετικές περιοχές. Δεν χρειάζεται έλεγχος των εισαγόμενων δεδομένων ως προς την εγκυρότητά τους.

Μονάδες 5

4.2 Θα υπολογίζει και θα εμφανίζει τη βαθμολογία κάθε διαμερίσματος, σύμφωνα με τα δοσμένα κριτήρια.

Μονάδες 10

4.3 Θα εντοπίζει το διαμέρισμα με τη μεγαλύτερη βαθμολογία. Θα εμφανίζει την περιοχή στην οποία βρίσκεται με κατάλληλο μήνυμα. Θεωρείστε ότι το διαμέρισμα αυτό είναι μοναδικό.

Μονάδες 10

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1  MAX_B = -1 # 4.3 Αρχικοποίηση της μέγιστης βαθμολογίας
2
3  # 4.1: εισαγωγή των δεδομένων
4  for i in range(7):
5      AREA = raw_input('Δώσε την περιοχή του διαμερίσματος:')
6      TIMH = float(input('Δώσε την τιμή ανά τ.μ. του διαμερίσματος σε ευρώ:'))
7      EMB = float(input('Δώσε την επιφάνεια του διαμερίσματος σε τ.μ.:'))
8      FLOOR= int(input('Δώσε τον όροφο του διαμερίσματος (0 για ισόγειο):'))
9
10 # 4.2: υπολογισμός βαθμολογίας διαμερίσματος
11     if EMB >= 120:
12         B= 2
13     else:
14         B= 0
15
16     if TIMH <= 6.0:
17         B= B + 3
18
19     if FLOOR >0 and FLOOR<=4:
20         B= B + 1
21     print 'Βαθμολογία διαμερίσματος:', B
22
23 # 4.3: αν η βαθμολογία του είναι μεγαλύτερη από το τρέχον μέγιστο
24 # ενημερώνεται η μέγιστη βαθμολογία και η περιοχή
25     if B > MAX_B:
26         MAX_B = B
27         MAX_AREA = AREA
28
29 # 4.3
30     print 'Περιοχή του διαμερίσματος με τη μεγαλύτερη βαθμολογία: ', MAX_AREA
```

ΘΕΜΑ 2

2.1 α) Να γράψετε το παρακάτω τμήμα προγράμματος σε Python, χρησιμοποιώντας την εντολή επανάληψης for αντί της εντολής επανάληψης while έτσι ώστε να εμφανίζει ακριβώς τις ίδιες τιμές.

```
1 j=5
2 while j <=17:
3     print j,
4     j=j+3
```

β) Να γράψετε στο γραπτό σας τις τιμές που εμφανίζονται στην οθόνη κατά την εκτέλεση του παραπάνω τμήματος προγράμματος.

Μονάδες 11

2.2 Δίνεται το παρακάτω τμήμα προγράμματος σε γλώσσα προγραμματισμού Python, το οποίο υπολογίζει και εμφανίζει στην οθόνη το άθροισμα των ακεραίων αριθμών που είναι πολλαπλάσιοι του 7 και βρίσκονται στο διάστημα από το 1 έως το 100.

```
s= (1)
for x in range(1, (2), 1):
    if x (3) 7== (4) :
        s= (5) + (6)
print "Το άθροισμα ισούται με: ", (7)
```

Στο πρόγραμμα υπάρχουν επτά υπογραμμισμένα αριθμημένα κενά. Να γράψετε τους αριθμούς 1, 2, 3, 4, 5, 6 και 7 που αντιστοιχούν στα κενά του παραπάνω προγράμματος και δίπλα σε κάθε αριθμό αυτό που πρέπει να συμπληρωθεί για να υλοποιείται ορθά το τμήμα προγράμματος.

Μονάδες 14

ΘΕΜΑ 2

2.1

α)

```
1  for j in range (5, 18, 3) :  
2      print j,
```

β. 5 8 11 14 17

2.2

1. 0
2. 101
3. %
4. 0
5. s
6. x
7. s

Εναλλακτικά:

5. x
6. s

ΘΕΜΑ 4

Μία μαθήτρια θέλει να αγοράσει ένα smartphone αξίας 300 ευρώ. Για τον σκοπό αυτό αποφάσισε να αποταμιεύσει τα χρήματα από το ημερήσιο χαρτζιλίκι που τις δίνουν οι γονείς της. Να αναπτύξετε πρόγραμμα σε γλώσσα προγραμματισμού Python, το οποίο να υλοποιεί τα παρακάτω:

4.1 Να διαβάζει κάθε μέρα, το ημερήσιο ποσό που της δίνουν οι γονείς της μέχρι το συνολικό ποσό αποταμίευσης από τα χαρτζιλίκια, να γίνει ίσο ή μεγαλύτερο από τα 300 ευρώ. Θεωρήστε ότι δεν υπάρχει ημέρα χωρίς χαρτζιλίκι.

Μονάδες 13

4.2 Να υπολογίζει και να εμφανίζει το πλήθος των ημερών που χρειάστηκε για τον σκοπό αυτό.

Μονάδες 5

4.3 Να υπολογίζει και να εμφανίζει πιθανό περίσσευμα χρημάτων. Στη συνέχεια, να εμφανίζει κατάλληλο μήνυμα, αν το περίσσευμα επαρκεί για την αγορά μιας θήκης smartphone αξίας 12 ευρώ.

Μονάδες 7

Σημείωση: Δεν απαιτούνται έλεγχοι ορθότητας δεδομένων.

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
    print b
```

ΘΕΜΑ 4

```
1  sf = 0 # αρχικοποιήσεις
2  d = 0
3  e = 0
4  pos = 0
5  # 4.1
6  while sf < 300 : # βρόχος επανάληψης
7      f = float(input("Δώσε το ημερήσιο χαρτζηλίκι: "))
8      sf = sf + f
9      d +=1
10
11  print 'Χρειάστηκαν ',d,' μέρες για να αποταμιευτεί το ποσό.' # 4.2
12  # 4.3
13  per = sf - 300
14  if per > 0:
15      print 'Υπάρχει περίσσειμα ',per,'€'
16      if per >= 12 : # εμφωλευμένη επιλογή
17          print 'Το περίσσειμα επαρκεί για την αγορά θήκης 12€'
```


ΘΕΜΑ 2

2.1 Δίνεται το εξής τμήμα προγράμματος σε γλώσσα προγραμματισμού Python.

```
1 for i in range(20,5,-5):  
2     for j in range(2,14,4):  
3         print i,
```

Να γράψετε τις τιμές που εμφανίζονται στην οθόνη κατά την εκτέλεσή του.

Μονάδες 9

2.2 Δίνεται το παρακάτω τμήμα προγράμματος σε γλώσσα προγραμματισμού Python όπου υπάρχουν τέσσερα (4) αριθμημένα κενά.

```
1 i= (1)  
2 sum=0  
3 while i <= (2) :  
4     sum=sum+ (3)  
5     i=i+ (4)
```

Να γράψετε στο τετράδιό σας τους αριθμούς (1), (2), (3), και (4) που αντιστοιχούν στα κενά του παραπάνω τμήματος προγράμματος και δίπλα σε κάθε αριθμό, αυτό που πρέπει να συμπληρωθεί, ώστε να υπολογιστούν τα παρακάτω αθροίσματα:

α) $30+50+70+90+110$

β) $2^3+4^3+6^3+8^3$

Μονάδες 16

ΘΕΜΑ 2

2.1 20 20 20 15 15 15 10 10 10

2.2

$\alpha)$

1. **30**

2. **110**

3. **i**

4. **20**

$\beta)$

1. **2**

2. **8**

3. **i**3**

4. **2**

ΘΕΜΑ 4

Στον διαδικτυακό μαθηματικό διαγωνισμό Καγκουρό επιτρέπεται από τους μαθητές/τριες Γυμνασίου η χρήση αριθμομηχανής για την εκτέλεση απλών αριθμητικών πράξεων. Να γραφεί πρόγραμμα σε γλώσσα προγραμματισμού Python το οποίο υλοποιεί τις παρακάτω λειτουργίες:

4.1 Να εμφανίζει το εξής μενού επιλογών:

1. Πρόσθεση
2. Αφαίρεση
3. Πολλαπλασιασμός
4. Διαίρεση
5. Ακέραιο υπόλοιπο
6. Δυνάμεις
7. Έξοδος

Στη συνέχεια, να διαβάζει από τον μαθητή/τρια ένα νούμερο από 1 μέχρι και 7. Απαιτείται να γίνεται έλεγχος εγκυρότητας δεδομένων.

Μονάδες 10

4.2 Εάν επιλεγεί κάποια από τις λειτουργίες 1-6, να εισάγονται από τον μαθητή/τρια δύο αριθμοί, να εκτελείται η αντίστοιχη πράξη και να εμφανίζεται στην οθόνη το αποτέλεσμα της.το αποτέλεσμα της.

Μονάδες 9

4.3 Στην περίπτωση της διαίρεσης και του ακέραιου υπολοίπου να ελέγχεται η διαίρεση με το μηδέν εμφανίζοντας κατάλληλο μήνυμα. Όταν επιλεγεί το νούμερο 7, το πρόγραμμα να εμφανίζει το μήνυμα «Τέλος».

Μονάδες 6

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
print b
```

ΘΕΜΑ 4

```
1 # 4.1
2 print "1. Πρόσθεση"
3 print "2. Αφαίρεση"
4 print "3. Πολλαπλασιασμός"
5 print "4. Διαίρεση"
6 print "5. Ακέραιο υπόλοιπο"
7 print "6. Δυνάμεις"
8 print "7. Έξοδος"
9 c=int(input("Παρακαλώ επιλέξτε νούμερο: "))
10 while not(c>=1 and c<=7): # έλεγχος εγκυρότητας
11     c=int(input("Προσοχή, επιλέξτε νούμερο από 1 μέχρι και 7: "))
12 # 4.2
13 if (c!=7):
14     x = float(input("Παρακαλώ πληκτρολόγησε τον πρώτο αριθμό: "))
15     y = float(input("Παρακαλώ πληκτρολόγησε τον δεύτερο αριθμό: "))
16     if c == 1:
17         print "x + y =", x+y
18     elif c == 2:
19         print "x - y =", x-y
20     elif c == 3:
21         print "x * y =", x*y
22     elif c == 4:
23         if y!=0: # εμφωλευμένη επιλογή
24             print "x / y =", x/y
25         else:
26             print "Δεν μπορεί να γίνει η διαίρεση"
27     elif c == 5:
28         if y!=0: # 4.3
29             print "x % y =", x%y
30         else:
31             print "Δεν μπορεί να γίνει η πράξη"
32     elif c == 6:
33         print "x ^ y =", x**y
34 else: # 4.3
35     print "Τέλος"
```

ΘΕΜΑ 2

2.1 Δίνεται το παρακάτω τμήμα προγράμματος σε γλώσσα προγραμματισμού Python:

```
1 for i in range (10, 200, 5):  
2     print i
```

Αυτό το τμήμα προγράμματος εμφανίζει διαδοχικά τους αριθμούς 10, 15, 20, ... , 195. Να τροποποιήσετε τον παραπάνω κώδικα έτσι ώστε οι αριθμοί αυτοί να εμφανίζονται στην οθόνη σε αντίστροφη διάταξη.

Μονάδες 9

2.2 Δίνεται το παρακάτω τμήμα προγράμματος σε γλώσσα προγραμματισμού Python:

```
1 i=2  
2 p=1  
3 pr=1  
4 while i<10:  
5     if i<=6:  
6         p=2  
7     else:  
8         p=1  
9     pr=pr*p  
10    print pr  
11    i=i+2
```

Να γράψτε όλες τις τιμές οι οποίες εμφανίζονται στην οθόνη.

Μονάδες 16

ΘΕΜΑ 2

2.1

```
1 for i in range (195, 5, -5):  
2     print i
```

2.2

2

4

8

8

ΘΕΜΑ 4

Ένας σινεφίλ, που του αρέσει να βλέπει πολλές κινηματογραφικές ταινίες, εγγράφεται συνδρομητής σε διαδικτυακό κανάλι ταινιών με σκοπό να αγοράσει/κατεβάσει ταινίες. Διαθέτει το ποσό των 80 ευρώ. Να αναπτύξετε πρόγραμμα σε Python το οποίο:

4.1 Να διαβάζει για κάθε ταινία, την τιμή και τη χώρα προέλευσής της (“Ε” για ελληνική και “Μ” για μεταγλωττισμένη) και να επιτρέπει την αγορά της, εφόσον η τιμή της δεν υπερβαίνει το διαθέσιμο υπόλοιπο χρημάτων. Διαφορετικά, να τερματίζει εμφανίζοντας το μήνυμα στην οθόνη «Τερματισμός αγορών».

Μονάδες 10

4.2 Να εμφανίζει στην οθόνη, το πλήθος των ελληνικών ταινιών και το πλήθος των μεταγλωττισμένων ταινιών που αγόρασε/κατέβασε ο συνδρομητής.

Μονάδες 5

4.3 Να εμφανίζει στην οθόνη το μεγαλύτερο ποσό που ξόδεψε για την αγορά μιας ταινίας.

Μονάδες 5

4.4 Να εμφανίζει στην οθόνη, το ποσό που περίσσεψε, εφόσον αυτό υπάρχει.

Μονάδες 5

Σημείωση. Για να είναι εμφανής η ύπαρξη της ίδιας εσοχής, συστήνεται οι μαθητές στο γραπτό τους να σημειώνουν τις εντολές που ανήκουν στο ίδιο μπλοκ με μία κάθετη γραμμή μπροστά από αυτές. Παράδειγμα:

```
while x<4:
    b=b+1
    a=a*x
    print a, x
    if b%2 == 0:
        x=x+1
        print x
    print b
```


ΘΕΜΑ 4

```
1  # Αρχικοποίηση μεταβλητών
2  poso=80
3  total=0
4  total_greek=0
5  max=0
6  synolo=0
7  # 4.1
8  agora = float(input("Τιμή αγοράς ταινίας: "))
9  # Έλεγχος αν η τιμή δεν υπερβαίνει το διαθέσιμο υπόλοιπο χρημάτων
10 while poso-agora>=0:
11     # Πλήθος ταινιών
12     total+=1
13     synolo=synolo+agora
14     print "Μπορείτε να αγοράσετε την ταινία"
15     poso=poso-agora
16     xora=raw_input("Εισάγετε χώρα προέλευσης: ")
17     # Έλεγχος χώρας Ε ή Μ
18     if xora=='Ε':
19         # 4.2
20         total_greek+=1
21     # 4.3 Εύρεση μεγαλύτερου ποσού
22     if agora>max:
23         max=agora
24     agora = int(raw_input("Τιμή αγοράς νέας ταινίας: "))
25 print "Τερματισμός αγορών"
26 print "Το πλήθος των ελληνικών ταινιών είναι: ", total_greek
27 print "Το πλήθος των μεταγλωττισμένων ταινιών είναι: ", total-total_gre
28 print "Η ακριβότερη ταινία είχε τιμή: " , max
29 # 4.4
30 if 80-synolo>0:
31     print "Έχει περισσέψει το ποσό: ", 80-synolo
```