

ΠΑΝΕΛΛΑΔΙΚΕΣ ΕΞΕΤΑΣΕΙΣ ΕΠΑ.Λ. · Τρίτη 13 Ιουνίου 2017

# Προγραμματισμός Υπολογιστών

Ενδεικτικές απαντήσεις σε γλώσσα προγραμματισμού Python 2.7

ΕΤΟΣ

**2017**

Οι απαντήσεις είναι **ενδεικτικές**: κάθε επιστημονικά τεκμηριωμένη λύση είναι αποδεκτή. Ο κώδικας ακολουθεί αυστηρά τη σύνταξη της **Python 2.7**, όπως διδάσκεται στο μάθημα — η `print` ως εντολή, `raw_input()` για συμβολοσειρές, `input()` για αριθμούς και ακέραια διαίρεση μεταξύ ακεραίων.

## ΘΕΜΑ Α

### A1

| Πρόταση | Απάντηση | Αιτιολόγηση                                                                                                                                                                            |
|---------|----------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| α.      | Σωστό    | Η <code>while</code> εκτελείται όσο ισχύει μια συνθήκη, άρα καλύπτει και τις περιπτώσεις όπου το πλήθος των επαναλήψεων δεν είναι γνωστό εκ των προτέρων.                              |
| β.      | Λάθος    | Η <code>L.pop()</code> αφαιρεί και επιστρέφει το τελευταίο στοιχείο της λίστας. Η προσθήκη στο τέλος γίνεται με <code>L.append()</code> .                                              |
| γ.      | Σωστό    | Ο συμβολισμός <code>28.2E-5</code> είναι η εκθετική μορφή του <code>0.000282</code> , δηλαδή αριθμός κινητής υποδιαστολής ( <code>float</code> ).                                      |
| δ.      | Σωστό    | Η <code>int(x)</code> μετατρέπει την αριθμητική τιμή <code>x</code> σε ακέραιο, αποκόπτοντας το δεκαδικό μέρος.                                                                        |
| ε.      | Λάθος    | Το αντικείμενο δημιουργείται από τον κατασκευαστή ( <code>constructor</code> ), τη μέθοδο <code>__init__</code> . Ο αποδομητής ( <code>destructor</code> ) καταστρέφει το αντικείμενο. |

### A2

| ΣΤΗΛΗ Α                 | →  | ΣΤΗΛΗ Β                                         |
|-------------------------|----|-------------------------------------------------|
| 1. <code>str()</code>   | στ | Συνάρτηση μετατροπής μιας τιμής σε συμβολοσειρά |
| 2. <code>True</code>    | γ  | Λογική τιμή                                     |
| 3. <code>"False"</code> | δ  | Συμβολοσειρά (είναι σε εισαγωγικά)              |
| 4. <code>or</code>      | α  | Λογικός τελεστής                                |
| 5. <code>==</code>      | β  | Συγκριτικός τελεστής                            |

**Σημείωση:** Περισεύει το γράμμα ε (Αριθμητικός τελεστής).

## A3

| Εντολή                               | Αποτέλεσμα               |
|--------------------------------------|--------------------------|
| <code>range(2,10)</code>             | [2, 3, 4, 5, 6, 7, 8, 9] |
| <code>range(2,10,3)</code>           | [2, 5, 8]                |
| <code>a = "abc" / print a*2</code>   | abcbac                   |
| <code>x=2 / y=3 / print 2*x+y</code> | 7                        |
| <code>a = 2 / print a**3</code>      | 8                        |

## A4

Για  $x = 3$  και  $y = 1$ :

| Λογική έκφραση                       | Ανάλυση                                    | Τιμή  |
|--------------------------------------|--------------------------------------------|-------|
| <code>not(x&gt;y)</code>             | $3 > 1 \rightarrow \text{True}$ , not True | False |
| <code>(x&gt;5) or (y&lt;2)</code>    | False or True                              | True  |
| <code>(x!=5) and (y!=0)</code>       | True and True                              | True  |
| <code>(x&lt;y) or (x**2&gt;y)</code> | False or $(9 > 1)$                         | True  |
| <code>x&lt;len("abc")</code>         | $3 < 3$                                    | False |

## ΘΕΜΑ Β

## B1

Η ταξινόμηση είναι φθίνουσα, οπότε σε κάθε πέρασμα «ανεβαίνει» προς την αρχή το μεγαλύτερο στοιχείο:

| Κενό | Συμπλήρωση |
|------|------------|
| (1)  | N          |
| (2)  | 1          |
| (3)  | i-1        |
| (4)  | j          |
| (5)  | >          |

Το συμπληρωμένο τμήμα προγράμματος:

```
python 2.7
```

```
N = len(lista)
for i in range(1, N, 1):
    for j in range(N-1, i-1, -1):
        if lista[j] > lista[j-1]:
            lista[j], lista[j-1] = lista[j-1], lista[j]
```

**B2**

Παρακολούθηση της εκτέλεσης ( $a=2$ ,  $x=1$ ,  $b=0$ ):

| Επανάληψη      | b | a         | Εμφανίζεται     | b%2                         | νέο x         |
|----------------|---|-----------|-----------------|-----------------------------|---------------|
| 1η ( $x=1<4$ ) | 1 | $2*1 = 2$ | $a = 2$ $x = 1$ | $1 \rightarrow \text{else}$ | $x = 1+2 = 3$ |
| 2η ( $x=3<4$ ) | 2 | $2*3 = 6$ | $a = 6$ $x = 3$ | $0 \rightarrow \text{if}$   | $x = 3+1 = 4$ |
| 3η ( $x=4$ )   | - | -         | τερματισμός     | -                           | -             |

Άρα στην οθόνη εμφανίζεται:

Έξοδος στην οθόνη

```
a= 2 x= 1
a= 6 x= 3
b= 2
```

**B3**

python 2.7

```
def SYN(a):
    if a < 10:
        return 2*a
    else:
        return 3*a
```

**B4**

H range(1,10,2) δίνει τις τιμές 1, 3, 5, 7, 9. Με while:

python 2.7

```
s = 0
i = 1
while i < 10:
    s = s + i
    print s
    i = i + 2
print s
```

**ΘΕΜΑ Γ****Γ1 - Γ4**

Πλήρες πρόγραμμα για το ΚΤΕΟ:

python 2.7

```
plithosF = 0
plithosE = 0
plithosM = 0
esodaF = 0
esodaE = 0
esodaM = 0
meli = 0
mi_meli = 0

# --- Γ1: εισαγωγή στοιχείων μέχρι τη λέξη TELOS ---
typos = raw_input("Δώσε τύπο οχήματος (F/E/M) ή TELOS: ")
while typos != "TELOS":
    melos = raw_input("Είναι μέλος του ΚΤΕΟ; (NAI/OXI): ")

    if melos == "NAI":
        meli = meli + 1
    else:
        mi_meli = mi_meli + 1

    if typos == "F":
        plithosF = plithosF + 1
        if melos == "NAI":
            esodaF = esodaF + 70
        else:
            esodaF = esodaF + 80
    elif typos == "E":
        plithosE = plithosE + 1
        if melos == "NAI":
            esodaE = esodaE + 40
        else:
            esodaE = esodaE + 50
    else:
        plithosM = plithosM + 1
        if melos == "NAI":
            esodaM = esodaM + 25
        else:
            esodaM = esodaM + 30

    typos = raw_input("Δώσε τύπο οχήματος (F/E/M) ή TELOS: ")

# --- Γ2: πλήθος και εισπράξεις ανά τύπο οχήματος ---
print "Φορτηγά", plithosF, esodaF, "ευρώ"
print "Επιβατικά", plithosE, esodaE, "ευρώ"
print "Μοτοσυκλέτες", plithosM, esodaM, "ευρώ"

# --- Γ3: συνολικό πλήθος και συνολικές εισπράξεις ---
synolo_oximatou = plithosF + plithosE + plithosM
synola_esoda = esodaF + esodaE + esodaM
print "Συνολικό πλήθος οχημάτων:", synolo_oximatou
print "Συνολικές εισπράξεις:", synola_esoda, "ευρώ"

# --- Γ4: μέλη και μη μέλη ---
print "Μέλη ΚΤΕΟ:", meli
print "Μη μέλη ΚΤΕΟ:", mi_meli
```

## ΘΕΜΑ Δ

## Δ1 - Δ4

python 2.7

```
NAME = []
VATHMOS = []

# --- Δ1: εισαγωγή με έλεγχο ορθότητας (1 έως 100) ---
for i in range(50):
    onoma = raw_input("Ονοματεπώνυμο μαθητή: ")
    NAME.append(onoma)

    vathmos = input("Βαθμολογία (1-100): ")
    while vathmos < 1 or vathmos > 100:
        vathmos = input("Λάθος τιμή. Βαθμολογία (1-100): ")
    VATHMOS.append(vathmos)

# --- Δ2: μέσος όρος ---
athroisma = 0
for i in range(50):
    athroisma = athroisma + VATHMOS[i]
MO = athroisma / 50.0
print "Μέσος όρος βαθμολογίας:", MO

# --- Δ3: μαθητές με βαθμολογία >= MO ---
print "Μαθητές με βαθμολογία μεγαλύτερη ή ίση του MO:"
for i in range(50):
    if VATHMOS[i] >= MO:
        print NAME[i], VATHMOS[i]

# --- Δ4: υψηλότερη βαθμολογία και όσοι την έχουν ---
megisto = VATHMOS[0]
for i in range(1, 50):
    if VATHMOS[i] > megisto:
        megisto = VATHMOS[i]

print "Υψηλότερη βαθμολογία:", megisto
print "Την πέτυχαν οι μαθητές:"
for i in range(50):
    if VATHMOS[i] == megisto:
        print NAME[i]
```

**Σημείωση:** Στο Δ2 η διαίρεση γίνεται με 50.0 (και όχι με 50), ώστε στην Python 2.7 να προκύψει πραγματικός και όχι ακέραιος μέσος όρος.