

ΠΑΝΕΛΛΑΔΙΚΕΣ ΕΞΕΤΑΣΕΙΣ ΕΠΑ.Λ. · Σάββατο 11 Ιουνίου 2022

Προγραμματισμός Υπολογιστών

Ενδεικτικές απαντήσεις σε γλώσσα προγραμματισμού Python 2.7

ΕΤΟΣ

2022

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.	Λάθος	Η στοίβα λειτουργεί ως LIFO (Last In First Out). Η ουρά είναι αυτή που λειτουργεί ως FIFO.
β.	Σωστό	Ο αλγόριθμος στηρίζεται στη διάταξη των στοιχείων, άρα προϋποθέτει ταξινομημένη συλλογή.
γ.	Σωστό	Οι τοπικές μεταβλητές δεσμεύονται με την κλήση και αποδεσμεύονται με τον τερματισμό της συνάρτησης.
δ.	Λάθος	Οι λίστες είναι δυναμική δομή: μπορούμε να προσθέτουμε και να αφαιρούμε στοιχεία.
ε.	Σωστό	Δεν υπάρχει <code>x</code> που να είναι ταυτόχρονα μικρότερο του 1 και μεγαλύτερο του 20, άρα η σύζευξη είναι πάντα False.

A2

ΣΤΗΛΗ Α	→	ΣΤΗΛΗ Β
1. Σχισιακός τελεστής	ε	!=
2. Αριθμητικός τελεστής	στ	+
3. Τελεστής λογικών πράξεων	γ	or
4. Τελεστής διαμέρισης	α	:
5. Υπαρξιακός τελεστής	β	in

Σημείωση: Περισσεύει το γράμμα δ (def), που είναι δεσμευμένη λέξη και όχι τελεστής.

ΘΕΜΑ Β

B1

Εντολή	Ανάλυση	Εμφανίζεται
<code>print L1[2]</code>	3ο στοιχείο της L1	ΔΩΔΕΚΑΝΗΣΑ
<code>print L2[-3]</code>	3ο από το τέλος της L2	ΝΑΞΟΣ
<code>print L1[1]+L2[1]</code>	συνένωση συμβολοσειρών	ΚΥΚΛΑΔΕΣΝΑΞΟΣ
<code>print len(L2)</code>	πλήθος στοιχείων της L2	4

Έξοδος στην οθόνη

```
ΔΩΔΕΚΑΝΗΣΑ
ΝΑΞΟΣ
ΚΥΚΛΑΔΕΣΝΑΞΟΣ
4
```

B2

α) Η μέθοδος `tipose()`:

```
python 2.7

class Mathitis:
    def __init__(self, am, onoma, vathmos):
        self.am = am
        self.onoma = onoma
        self.vathmos = vathmos

    def tipose(self):
        if self.vathmos >= 10:
            print "Προάγεται"
        else:
            print "Παραπέμπεται"
```

β) Δημιουργία των δύο αντικειμένων:

```
python 2.7

mathitis1 = Mathitis(103, "Νικολάου", 19)
mathitis2 = Mathitis(105, "Γεωργίου", 9)
```

γ) Κλήση της μεθόδου για το `mathitis1` (εμφανίζει «Προάγεται»):

```
python 2.7

mathitis1.tipose()
```

δ) Μέσος όρος των βαθμών των δύο αντικειμένων:

```
python 2.7

mo = (mathitis1.vathmos + mathitis2.vathmos) / 2.0
print "Μέσος όρος:", mo
```

Σημείωση: Η διαίρεση με 2.0 δίνει 14.0. Αν γραφόταν /2 στην Python 2.7, το αποτέλεσμα θα ήταν ο ακέραιος 14.

B3

Κενό	(1)	(2)	(3)	(4)	(5)	(6)
Συμπλήρωση	0	word	letter	m	1	m

Η συμπληρωμένη συνάρτηση:

```
python 2.7

def plithos_A(word):
    m = 0
    for letter in word:
        if letter == 'A':
            m = m + 1
    return m
```

ΘΕΜΑ Γ

Γ1 - Γ3

```
python 2.7

# --- Γ3: κλιμακωτός υπολογισμός της χρέωσης ---
def ypologismos(temaxia):
    if temaxia <= 3:
        xrewsi = temaxia * 120
    elif temaxia <= 6:
        xrewsi = 3 * 120 + (temaxia - 3) * 100
    else:
        xrewsi = 3 * 120 + 3 * 100 + (temaxia - 6) * 70
    return xrewsi

esoda = 0
plithos_pano10 = 0

# --- Γ1: για κάθε έναν από τους 50 πελάτες ---
for i in range(50):
    temaxia = input("Αριθμός τεμαχίων: ")
    xrewsi = ypologismos(temaxia)
    print "Χρέωση πελάτη:", xrewsi, "ευρώ"

    esoda = esoda + xrewsi
    if temaxia > 10:
        plithos_pano10 = plithos_pano10 + 1

# --- Γ2: συνολικά έσοδα και ποσοστό ---
print "Συνολικά έσοδα καταστήματος:", esoda, "ευρώ"
print "Ποσοστό πελατών με πάνω από 10 τεμάχια:", plithos_pano10 * 100.0 / 50, "%"
```

Σημείωση: Η χρέωση είναι κλιμακωτή: τα 3 πρώτα τεμάχια χρεώνονται 120 ευρώ το καθένα, τα επόμενα 3 από 100 ευρώ και όσα περισσεύουν πάνω από 6 από 70 ευρώ. Για παράδειγμα, για 5 τεμάχια η χρέωση είναι $3 \cdot 120 + 2 \cdot 100 = 560$ ευρώ.

ΘΕΜΑ Δ**Δ1 - Δ4**

```
python 2.7

KOD = []
VATH = []

# --- Δ1: εισαγωγή στοιχείων μέχρι τη λέξη ΤΕΛΟΣ ---
kodikos = raw_input("Κωδικός υποψηφίου (ΤΕΛΟΣ για τερματισμό): ")
while kodikos != "ΤΕΛΟΣ":
    KOD.append(kodikos)

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

    kodikos = raw_input("Κωδικός υποψηφίου (ΤΕΛΟΣ για τερματισμό): ")

N = len(VATH)

# --- Δ2: μέσος όρος βαθμολογίας ---
athroisma = 0
for i in range(N):
    athroisma = athroisma + VATH[i]
print "Μέσος όρος βαθμολογίας:", athroisma / float(N)

# --- Δ3: μεγαλύτερη βαθμολογία και οι κωδικοί που την έχουν ---
megisti = VATH[0]
for i in range(1, N):
    if VATH[i] > megisti:
        megisti = VATH[i]

print "Μεγαλύτερη βαθμολογία:", megisti
print "Κωδικοί υποψηφίων με αυτή τη βαθμολογία:"
for i in range(N):
    if VATH[i] == megisti:
        print KOD[i]

# --- Δ4: αρχείο με τους επιτυχόντες ---
arxeio = open("epityxon.txt", "w")
aa = 0
for i in range(N):
    if VATH[i] > 60:
        aa = aa + 1
        arxeio.write(str(aa) + ". " + KOD[i] + "\n")
arxeio.close()
```

Σημείωση: Ο αύξων αριθμός aa αυξάνεται μόνο για τους επιτυχόντες, ώστε η αρίθμηση των γραμμών του αρχείου να είναι συνεχόμενη.