

ΠΑΝΕΛΛΑΔΙΚΕΣ ΕΞΕΤΑΣΕΙΣ ΕΠΑ.Λ. · Τρίτη 11 Ιουνίου 2024

Προγραμματισμός Υπολογιστών

Ενδεικτικές απαντήσεις σε γλώσσα προγραμματισμού Python 2.7

ΕΤΟΣ

2024

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 μέθοδος καλείται μέσω του αντικειμένου (<code>antikeimeno.methodos()</code>), ενώ η συνάρτηση απλώς με το όνομά της.
β.	Λάθος	Στην ουρά (FIFO) οι εισαγωγές γίνονται από το ένα άκρο και οι εξαγωγές από το άλλο. Από ένα μόνο άκρο λειτουργεί η στοίβα.
γ.	Λάθος	Η Python είναι γλώσσα δυναμικών τύπων: η ίδια μεταβλητή μπορεί να πάρει τιμές διαφορετικού τύπου.
δ.	Λάθος	Η έκφραση <code>x and not x</code> είναι πάντα <code>False</code> , αφού ένα από τα δύο σκέλη είναι πάντα ψευδές.
ε.	Σωστό	Η εμβέλεια μιας τοπικής μεταβλητής περιορίζεται στη συνάρτηση όπου δηλώθηκε.

A2

ΣΤΗΛΗ Α	→	ΣΤΗΛΗ Β
1. <code>int("5")**2</code>	β	25
2. <code>str(5)*2</code>	στ	"55" (επανάληψη συμβολοσειράς)
3. <code>"5500"[:1]</code>	δ	"5"
4. <code>pow(5,0)</code>	γ	1
5. <code>divmod(25,5)</code>	α	(5,0) → πηλίκο 5, υπόλοιπο 0

Σημείωση: Περισσεύει το γράμμα ε (0,5).

ΘΕΜΑ Β

B1

python 2.7

```
def trim_a(s1):  
    s2 = ""  
    for letter in s1:  
        if letter != 'a' and letter != 'A':  
            s2 = s2 + letter  
    return s2
```

Σημείωση: Για $s1 = \text{"Alabama"}$ η συνάρτηση επιστρέφει "lbm" , όπως ζητά η εκφώνηση.

B2

Η λίστα είναι διατεταγμένη σε ΦΘΙΝΟΥΣΑ σειρά, άρα όταν το μεσαίο στοιχείο είναι μικρότερο από το ζητούμενο, η αναζήτηση συνεχίζεται στο αριστερό μισό.

α) Αναζήτηση του 85:

Βήμα	first	last	mid	A[mid]	Σύγκριση
1	0	9	4	73	$73 < 85 \rightarrow \text{last} = 3$
2	0	3	1	181	$181 > 85 \rightarrow \text{first} = 2$
3	2	3	2	145	$145 > 85 \rightarrow \text{first} = 3$
4	3	3	3	98	$98 > 85 \rightarrow \text{first} = 4 > \text{last} \rightarrow \text{δεν βρέθηκε}$

Συγκρίνονται οι αριθμοί: 73, 181, 145, 98

β) Αναζήτηση του 12:

Βήμα	first	last	mid	A[mid]	Σύγκριση
1	0	9	4	73	$73 > 12 \rightarrow \text{first} = 5$
2	5	9	7	29	$29 > 12 \rightarrow \text{first} = 8$
3	8	9	8	12	$12 == 12 \rightarrow \text{βρέθηκε}$

Συγκρίνονται οι αριθμοί: 73, 29, 12

B3

Η $\text{range}(10)$ δίνει $i = 0..9$ και η $\text{range}(10, -1, -1)$ δίνει $j = 10..0$:

python 2.7

```
i = 0
while i < 10:
    j = 10
    while j >= 0:
        print i*j
        j = j - 1
    i = i + 1
```

ΘΕΜΑ Γ

Γ1 - Γ5

python 2.7

```
plithos = 0
prokrithentes = 0
megisti = 0

# --- Γ1: εισαγωγή στοιχείων μέχρι τη λέξη ΤΕΛΟΣ ---
onoma = raw_input("Ονοματεπώνυμο (ΤΕΛΟΣ για τερματισμό): ")
while onoma != "ΤΕΛΟΣ":
    athroisma = 0
    ola_pano_12 = True

    for i in range(10):
        vathmos = input("Βαθμός θέματος (1-20): ")
        while vathmos < 1 or vathmos > 20:
            vathmos = input("Λάθος τιμή. Βαθμός θέματος (1-20): ")
        athroisma = athroisma + vathmos
        if vathmos < 12:
            ola_pano_12 = False

    # --- Γ2: τελική βαθμολογία ---
    teliki = athroisma / 10.0
    print "Τελική βαθμολογία:", teliki

    # --- Γ3: έλεγχος πρόκρισης ---
    if teliki > 15 and ola_pano_12:
        print "Ο/Η", onoma, "προκρίνεται στην επόμενη φάση"
        prokrithentes = prokrithentes + 1

    # --- Γ4: εντοπισμός μεγαλύτερης τελικής βαθμολογίας ---
    plithos = plithos + 1
    if plithos == 1 or teliki > megisti:
        megisti = teliki

    onoma = raw_input("Ονοματεπώνυμο (ΤΕΛΟΣ για τερματισμό): ")

# --- Γ4 ---
print "Μεγαλύτερη τελική βαθμολογία:", megisti

# --- Γ5 ---
print "Ποσοστό διαγωνιζομένων που προκρίθηκαν:",
print prokrithentes * 100.0 / plithos, "%"
```

Σημείωση: Η λογική μεταβλητή ola_rapo_12 αρχικοποιείται σε True για κάθε νέο διαγωνιζόμενο και γίνεται False μόλις βρεθεί έστω ένας βαθμός μικρότερος του 12.

ΘΕΜΑ Δ

Δ1 - Δ4

```
python 2.7

# --- Δ1: ανάγνωση των ονομάτων από το αρχείο ---
ON = []
arxeio = open("branch.txt", "r")
grammi = arxeio.readline()
while grammi != "":
    ON.append(grammi.strip())
    grammi = arxeio.readline()
arxeio.close()

N = len(ON)

# --- Δ2: εισπράξεις Ιουνίου για κάθε υποκατάστημα ---
S_POSO = []
for i in range(N):
    athroisma = 0
    for j in range(30):
        eispraxi = input("Ημερήσια είσπραξη: ")
        athroisma = athroisma + eispraxi
    S_POSO.append(athroisma)

# --- Δ3: μέσος όρος και πλήθος υποκαταστημάτων >= M0 ---
synolo = 0
for i in range(N):
    synolo = synolo + S_POSO[i]
M0 = synolo / float(N)
print "Μέσος όρος εισπράξεων:", M0

plithos = 0
for i in range(N):
    if S_POSO[i] >= M0:
        plithos = plithos + 1
print "Υποκαταστήματα με εισπράξεις >= M0:", plithos

# --- Δ4: ταξινόμηση φθίνουσα, αλφαβητικά σε ισότητα εισπράξεων ---
for i in range(N-1):
    for j in range(N-1, i, -1):
        if S_POSO[j] > S_POSO[j-1] or \
           (S_POSO[j] == S_POSO[j-1] and ON[j] < ON[j-1]):
            S_POSO[j], S_POSO[j-1] = S_POSO[j-1], S_POSO[j]
            ON[j], ON[j-1] = ON[j-1], ON[j]

for i in range(N):
    print ON[i], S_POSO[i]
```

Σημείωση: Η μέθοδος strip() αφαιρεί τον χαρακτήρα αλλαγής γραμμής από το τέλος κάθε γραμμής του αρχείου.