

PANELLINIES ONLINE · Σάββατο 1 Αυγούστου 2026

Προσομοίωση Πανελλαδικών Εξετάσεων

Προγραμματισμός Υπολογιστών (ΕΠΑ.Λ.) · Ενδεικτικές απαντήσεις ·
Διάρκεια: 180 λεπτά

ΜΟΝΑΔΕΣ

100

Πώς φτιάχτηκε. Τα ΘΕΜΑΤΑ Α και Β προέρχονται από τράπεζα υλικού βασισμένη στα πραγματικά θέματα των Πανελλαδικών 2017-2026, με τις απαντήσεις τους υπολογισμένες. Τα ΘΕΜΑΤΑ Γ και Δ γράφτηκαν από γλωσσικό μοντέλο (nvidia/nemotron-3-super-120b-a12b) με βάση τις σημειώσεις του μαθήματος. Δεν είναι επίσημο θέμα του Υπουργείου Παιδείας — χρησιμοποιήσέ το για εξάσκηση και διασταύρωσε τις λύσεις των ΘΕΜΑΤΩΝ Γ και Δ με τον καθηγητή σου.

ΘΕΜΑ Α

A1

Πρόταση	Απάντηση	Αιτιολόγηση
α.	Λάθος	Η int αποθηκεύει μόνο ακέραιους· ο δεκαδικός μέρος χάνεται αυτόματα.
β.	Σωστό	Η pop σε κενή στοίβα προκαλεί σφάλμα γιατί δεν υπάρχει στοιχείο να αφαιρεθεί.
γ.	Σωστό	Η κληρονομία επιτρέπει στην υποκλάση να παρέχει νέα υλοποίηση μεθόδου υπερκλάσης.
δ.	Σωστό	Η δυαδική αναζήτηση υποθέτει ότι τα δεδομένα είναι ταξινομημένα αλλιώς δεν εγγυάται σωστό αποτέλεσμα.
ε.	Λάθος	Ο while ελέγχει τη συνθήκη πριν την πρώτη εκτέλεση του σώματος· αν η συνθήκη είναι ψευδής το σώμα δεν εκτελείται ποτέ.

A2

Σημείωση: Το γράμμα ε (0,5) περισσεύει.

ΘΕΜΑ Β

B2

ΛΥΣΗ:

Επανάληψη	lexi	len(lexi)-1	result μετά τη συνένωση
1	"apple"	4	"e"
2	"banana"	5	"ea"
3	"cherry"	5	"eay"

Έξοδος στην οθόνη

eay

Σημείωση: Η μεταβλητή result αρχικοποιείται σε κενή συμβολοσειρά και σε κάθε επανάληψη προσαρμόζεται το τελευταίο γράμμα (θέση len(lexi)-1) της τρέχουσας λέξης.

B3

A) Να ξαναγράψετε το τμήμα χρησιμοποιώντας μόνο εντολές while, ώστε να εμφανίζεται το ίδιο αποτέλεσμα.

α) Μετατροπή του τμηματικού κώδικα (μον. 5)

Λύση:

python 2.7

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

Έξοδος στην οθόνη

```
0
0
0
0
0
0
0
0
0
0
0
10
9
8
7
6
5
4
```

Έξοδος στην οθόνη

3
2
1
0
20
18
16
14
12
10
8
6
4
2
0
30
27
24
21
18
15
12
9
6
3
0
40
36
32
28
24
20
16
12
8
4
0
50
45
40
35
30
25
20
15
10
5
0
60
54
48
42
36
30

Έξοδος στην οθόνη

24
18
12
6
0
70
63
56
49
42
35
28
21
14
7
0
80
72
64
56
48
40
32
24
16
8
0
90
81
72
63
54
45
36
27
18
9
0
100
90
80
70
60
50
40
30
20
10
0

ΘΕΜΑ Γ

Γ1

Προτεινόμενη λύση:

python 2.7

```
for i in range(5):
    onoma = raw_input("Δώσε όνομα πακέτου: ")
    typos = raw_input("Δώσε τύπο πακέτου (πολιτιστικό, περιπέτεια, χαλαρωτικό): ")
    plithos = input("Δώσε πλήθος συμμετεχόντων: ")
    while plithos < 10 or plithos > 30:
        print "Το πλήθος πρέπει να είναι από 10 έως 30."
        plithos = input("Δώσε πλήθος συμμετεχόντων: ")
```

Γ2

α) Καλείτε τη συνάρτηση YPOLOGISMOS(typos, plithos) και αποθηκεύετε την επιστροφή στην μεταβλητή cost. (μον. 3)

β) Εκτυπώνετε το αποτέλεσμα με φιλικό μήνυμα που περιλαμβάνει το όνομα του πακέτου και το κόστος σε ευρώ. (μον. 3)

python 2.7

```
cost = YPOLOGISMOS(typos, plithos)
print "Το κόστος του πακέτου", onoma, "είναι", cost, "ευρώ."
```

Γ3

python 2.7

```
def YPOLOGISMOS(typos, plithos):
    if typos == "πολιτιστικό":
        return plithos * 15
    elif typos == "περιπέτεια":
        return plithos * 20
    elif typos == "χαλαρωτικό":
        return plithos * 10
    else:
        return 0
```

Γ4

Προσθέτουμε μεταβλητές για το άθροισμα των εσόδων και για τον αριθμό των περιπετειακών πακέτων, και μετά το τέλος του βρόχου εκτυπώνουμε τα ζητούμενα.

python 2.7

```
synolo_esoda = 0
count_peripeteia = 0
synolo_paketa = 0
for i in range(5):
    onoma = raw_input("Δώσε όνομα πακέτου: ")
    typos = raw_input("Δώσε τύπο πακέτου (πολιτιστικό, περιπέτεια, χαλαρωτικό): ")
    plithos = input("Δώσε πλήθος συμμετεχόντων: ")
    while plithos < 10 or plithos > 30:
        print "Το πλήθος πρέπει να είναι από 10 έως 30."
        plithos = input("Δώσε πλήθος συμμετεχόντων: ")
    cost = YPOLOGISMOS(typos, plithos)
    print "Το κόστος του πακέτου", onoma, "είναι", cost, "ευρώ."
    synolo_esoda = synolo_esoda + cost
    synolo_paketa = synolo_paketa + 1
    if typos == "περιπέτεια":
        count_peripeteia = count_peripeteia + 1
print "Τα συνολικά έσοδα της εβδομάδας είναι", synolo_esoda, "ευρώ."
if synolo_paketa > 0:
    pososto = (count_peripeteia * 100.0) / synolo_paketa
else:
    pososto = 0
print "Το ποσοστό των περιπετειακών πακέτων είναι", pososto, "% pososto, %."
```

ΘΕΜΑ Δ

Δ1

Ορίζουμε τη λίστα ANS με τις σωστές απαντήσεις. Για κάθε υποψήφιο διαβάζουμε τον κωδικό και τον αποθηκεύουμε στη λίστα KOD. Διαβάζουμε τις δέκα απαντήσεις, τις συγκρίνουμε με τις αντίστοιχες στοιχεία του ANS και αυξάνουμε τη βαθμολογία για κάθε σωστή απάντηση. Τελικά την βαθμολογία την αποθηκεύουμε στη λίστα SB.

python 2.7

```
# Ορισμός της λίστας σωστής λίστας
ANS = ["soston apantiseon"]
ANS = ["A", "B", "C", "D", "E", "A", "B", "C", "D", "E"]

KOD = [] # λίστα κωδικών
SB = [] # λίστα βαθμολογιών

n = input("Δώσε τον αριθμό των υποψηφίων: ")
for _ in range(n):
    kod = raw_input("Κωδικός υποψηφίου: ")
    KOD.append(kod)
    score = 0
    for i in range(10):
        ans = raw_input("Απάντηση {}: ".format(i+1))
        if ans == ANS[i]:
            score += 1
    SB.append(score)

# Προαιρετική εμφάνιση αποτελεσμάτων για έλεγχο
print "Κωδικοί:", KOD
print "Βαθμοί:", SB
```

Δ2

...

python 2.7

...

Δ3

python 2.7

```
max_score = SB[0]
for i in range(1, len(SB)):
    if SB[i] > max_score:
        max_score = SB[i]
for i in range(len(SB)):
    if SB[i] == max_score:
        print KOD[i]
```

Δ4

Η λύση διαβάζει τις λίστες KOD (κωδικούς) και SB (βαθμούς) που έχουν ήδη δημιουργηθεί, ελέγχει για κάθε υποψήφιο αν η βαθμολογία του είναι ≥ 5 και, αν ναι, γράφει στο αρχείο τον τρέχοντα αριθμό γραμμής, τελεία και κωδικό. Τέτοια επανάληψη χρησιμοποιεί βρόχο for και την εντολή write για εγγραφή σε αρχείο.

python 2.7

```
fout = open("epityxon.txt", "w")
line_num = 1
for i in range(len(KOD)):
    if SB[i] >= 5:
        fout.write(str(line_num) + ". " + KOD[i] + "\n")
        line_num = line_num + 1
fout.close()
print "To αρχείο epityxon.txt δημιουργήθηκε."
```