



Quiz 3ο Κεφάλαιο Τυχαίες ερωτήσ



mood



Προγραμμα

Αρχική

Τα μαθήματά μου



> Τα μαθήματά μου > Python Γ

Ερώτηση **1**

Λανθασμένη

Βαθμολογήθηκε
0,00 στα 1,00



Σημαιοδότηση
ερώτησης

$5 * (3 + 2) / 10$

Οι πράξεις μέσο

Οι πράξεις μέσα στις παρενθέσεις έχουν ακόμα μεγαλύτερη προτεραιότητα

$$5 * (3 + 2) / 10 =$$

$$5 * 5 / 10 =$$

$$25 / 10 =$$

2

Η σωστή απάντηση είναι: 2

 Quiz 3ο Κεφάλαιο Τυχαίες ερωτήσεις ✕

← → ↻ 🏠 🔒 moodle.sep

🏠 Αρχική 👜 Τα μαθήματα

Ερώτηση **2**

Σωστό

Βαθμολογήθηκε
1,00 στα 1,00

🚩

Σημαιοδότηση
ερώτησης

2*ΕΠΑΛ'

Ο τελεστής *

Ερώτηση 2

Σωστό

Βαθμολογήθηκε
1,00 στα 1,00



Σημαιοδότηση
ερώτησης

2*ΕΠΑΛ'

ΕΠΑΛΕΠΑ



Ο τελεστής * όταν εφαρμόζεται σε αντικείμενα τύπου string με ακέραιο N έχει σαν αποτέλεσμα μια συμβολοσειρά που προκύπτει από N επαναλήψεις του string.

Η σωστή απάντηση είναι: ΕΠΑΛΕΠΑΛ

Ερώτηση **3**

Μη πλήρης

Βαθμολογήθηκε
στα 1,00



Σημαιοδότηση
ερώτησης

$11\%3-2*2$

Έλεγχος

Ερώτηση **4**

Μη πλήρης

Βαθμολογήθηκε
στα 1,00



Σημαιοδότηση
ερώτησης

$\text{pow}(3,2)$

Έλεγχος

Ερώτηση **3**

Σωστό

Βαθμολογήθηκε
1,00 στα 1,00



Σημαιοδότηση
ερώτησης

11%3-2*2

-2



11%3-2*2 =

2 - 4=

-2

Η σωστή απάντηση είναι: -2



pow(3,2) ----> 3²

Η σωστή απάντηση είναι: 9

ΕΡΩΤΗΣΗ 5

not False or False and False

Quiz 3ο Κεφάλαιο Τυχαιές ερωτήσεις X +

moodle.sepchious.gr/mod/quiz/attempt.php?attempt=14343&cmid=1157&page=1

Αρχική Τα μαθήματά μου Τρέχον μάθημα Σ.Ε.Π. Χίου Erasmus

Ερώτηση 6
Σωστό
Βαθμολογήθηκε
1,00 στα 1,00
Σημαιοδότηση
ερώτησης

not False or False and False

Επιλέξτε ένα:

☒ True 

☐ False

Η απάντησή σας είναι σωστή
Η προτεραιότητα των λογικών τελεστών είναι **not, and, or** με αυτή τη σειρά.
Αυτό σημαίνει ότι αν δεν υπάρχουν παρενθέσεις, πρώτα εκτελούνται τα **not** μετά τα **and** και μετά τα **or**
not False or False and False =
True or **False and False** =
True or False =
True

Type here to search

Ερώτηση 6

not (False or False) and False

Η προτεραιότητα των λογικών τελεστών είναι **not, and, or** με αυτή τη σειρά αν δεν υπάρχουν παρενθέσεις.

not (**False or False**) and False =

not False and False =

True and False =

False

Οι σωστή απάντηση είναι: False

Ερώτηση **7**

Μη πλήρης

Βαθμολογήθηκε
στα 1,00



Σημαιοδότηση
ερώτησης

$1 \geq 2$

Επιλέξτε ένα:

☐ True

☐ False

Έλεγχος

Η απάντησή σας είναι σωστή

Οι σωστή απάντηση είναι: False

Επιλέξτε τις εντολές Python που αντιστοιχούν στις παρακάτω αλγεβρικές παραστάσεις. **$a = x(y+z(5-k))$**

Επιλέξτε μία ή παραπάνω σωστές απαντήσεις.

- ☐ **$a = x*(y+z*(5-k))$**
- ☐ **$a = x(y+z(5-k))$**

Ερώτηση 8

Σωστό

Βαθμολογήθηκε
1,00 στα 1,00



Σημαιοδότηση
ερώτησης

Επιλέξτε τις εντολές Python που αντιστοιχούν στις παρακάτω αλγεβρικές παραστάσεις.

$$a = x(y+z(5-k))$$

Επιλέξτε μία ή παραπάνω σωστές απαντήσεις.

☒ `a = x*(y+z*(5-k))`

☐ `a = x(y+z(5-k))`



Η απαντησή σας είναι σωστή

Οι σωστή απάντηση είναι: `a = x*(y+z*(5-k))`

ΑΚΟΛΟΥΘΕΙ ΕΡΩΤΗΣΗ ΠΑΓΙΔΑ



Quiz 3ο Κεφάλαιο Τυχαίες ερωτήσεις



moodle.sepchou.gr/mod/quiz/attempt.php?attempt=14343&cmid=1157&page=2

🏠 Αρχική



Τα μαθήματά μου



Τρέχον μάθημα

Σ.Ε.Π. Χίου

Erasmus

Ερώτηση 9

Λανθασμένη

Βαθμολογήθηκε
0,00 στα 1,00



Σημαιοδότηση
ερώτησης

Επιλέξτε τις εντολές Python που αντιστοιχούν στις παρακάτω αλγεβρικές παραστάσεις.

$$y = a + \frac{1}{x}$$

Επιλέξτε μία ή παραπάνω σωστές απαντήσεις.

☐ `y = a+1.0/x`

☒ `y = a + 1/x`

☐ `y = a+1/float(x)`



Στην Python 2 για να μη γίνει η ακέραια διαίρεση πρέπει να εξασφαλίσουμε ότι ο αριθμητής ή/και ο παρονομαστής του κλάσματος είναι float.

`y = a+1.0/x`

`y = a+1/float(x)`

Οι σωστές απαντήσεις είναι: `y = a+1.0/x`, `y = a+1/float(x)`

Τι θα εμφανιστεί για κάθε ομάδα τιμών

```
if x > y:
    print "Άλφα"

elif x < y:
    print "Βήτα"

else:
    print "ΆλφαΒήτα"
```

x = 99

y = 99

Βήτα

Επιλέξτε...

Βήτα

ΆλφαΒήτα

Άλφα

x = 4

y = 8

x = 22

y = 2

Επιλέξτε...

Έλεγχος

Ερώτηση 9

Σωστό

Βαθμολογήθηκε
1,00 στα 1,00



Σημαιοδότηση
ερώτησης

Τι θα εμφανιστεί για κάθε ομάδα τιμών

```
if x > y:  
    print "Άλφα"  
  
elif x < y:  
    print "Βήτα"  
  
else:  
    print "ΆλφαΒήτα"
```

x = 99

y = 99

ΆλφαΒήτα ⇅



x = 4

y = 8

Βήτα ⇅



x = 22

y = 2

Άλφα ⇅



Ερώτηση 10

Μη πλήρης

Βαθμολογήθηκε
στα 1,00Σημαιοδότηση
ερώτησης

Αν $a = 1$ και $b = 2$, συμπληρώστε τις τιμές που θα έχουν τα a και b μετά την εκτέλεση των παρακάτω εντολών

 $a, b = b, a$

a

b

Έλεγχος

Ερώτηση **10**

Σωστό

Βαθμολογήθηκε
1,00 στα 1,00



Σημαιοδότηση
ερώτησης

Αν $a = 1$ και $b = 2$, συμπληρώστε τις τιμές που θα έχουν τα a και b μετά την εκτέλεση των παρακάτω εντολών

$a, b = b, a$

a

2



b

1



Αντιμετάθεση των τιμών δυο μεταβλητών χωρίς χρήση επιπλέον προσωρινής μεταβλητής.

A

G

Ερώτηση **12**

Μη πλήρης

Βαθμολογήθηκε
στα 1,00



Σημαιοδότηση
ερώτησης

Επιλέξτε την κατάλληλη λογική συνθήκη η οποία να αληθεύει όταν

Ο x είναι ίσος με 1 ή με 2 ή με 4

Επιλέξτε μία ή παραπάνω σωστές απαντήσεις.

☐ $x == 1$ and $x == 2$ and $x == 4$

☐ $x != 1$ or $x != 2$ or $x != 4$

☐ $x == 1$ or $x == 2$ or $x == 4$

☐ x in $[1, 2, 4]$

Έλεγχος

Εχετε επιλέξει σωστά 1.

Οι σωστές απαντήσεις είναι: $x==1$ or $x==2$ or $x==4$, $x \in [1,2,4]$

Εχετε επιλέξει σωστά 1.

Οι σωστές απαντήσεις είναι: $x==1$ or $x==2$ or $x==4$, $x \in [1,2,4]$

Να συμπληρώσετε κατάλληλα τα κενά, ώστε το παρακάτω πρόγραμμα σε Python να εμφανίζει ένα τρίγωνο όπως παρακάτω

*

**

Πρόγραμμα τρίγωνο

for i in range(1,):

for j in range(1,):

print ' ',

print

Έλεγχος

```
for i in range(1,[6]):  
    for j in range(1,[i+1]):  
        print ' *',  
    print
```

ΠΑΓΙΔΑ ΔΙΑΒΑΣΤΕ ΠΡΟΣΕΚΤΙΚΑ ΠΛΗΘΟΣ!!!!

Αρχική

Τα μαθήματά μου

Τρέχον μάθημα

Σ.Ε.Π. Χίου

Erasmus

Μη πλήρης
Βαθμολογήθηκε
στα 1,00
Σημαιοδότηση
ερώτησης

A4. Δίνεται το παρακάτω τμήμα προγράμματος σε γλώσσα προγραμματισμού Python:

```
for i in range(1,x):  
    for j in range(1,3):  
        print "*"
```

Να μεταφέρετε στο τετράδιό σας και να συμπληρώσετε τον παρακάτω πίνακα με το πλήθος των συμβόλων "*" που θα εμφανιστούν σε κάθε εκτέλεση του παραπάνω τμήματος προγράμματος για καθεμιά από τις τιμές του x:

x	Πλήθος συμβόλων "*"
2	
3	
4	

Μονάδες 6

x	Πλήθος συμβόλων "*"
2	-
3	
4	

Έλεγχος

ΑΠΑΝΤΗΣΗ

2

4

6

 Αρχική

 Τα μαθήματά μου

 Τρέχον μάθημα

Σ.Ε.Π. Χίου

Erasmus

Ερώτηση **15**

Μη πλήρης

Βαθμολογήθηκε
στα 1,00



Σημαιοδότηση
ερώτησης

Τι θα εμφανιστεί κατά την εκτέλεση του παρακάτω προγράμματος, αν δοθούν από το πληκτρολόγιο οι τιμές:

a=10 και b=5

Σημείωση:Μόνο ένα κενό ανάμεσα στις τιμές.

```
a,b= input(),input()
```

```
a=a-b
```

```
b=a+b
```

```
c=b
```

```
if a>c:
```

```
    c=a
```

```
print a,b,c
```

Ερώτηση **15**

Σωστό

Βαθμολογήθηκε
1,00 στα 1,00Σημαιοδότηση
ερώτησης

Τι θα εμφανιστεί κατά την εκτέλεση του παρακάτω προγράμματος, αν δοθούν από το πληκτρολόγιο οι τιμές:

a=10 και b=5

Σημείωση: Μόνο ένα κενό ανάμεσα στις τιμές.

```
a,b= input(),input()
```

```
a=a-b
```

```
b=a+b
```

```
c=b
```

```
if a>c:
```

```
    c=a
```

```
print a,b,c
```

Απάντηση: 5 10 10



```
num=int(input("Δώστε αριθμό"))  
check=num%2  
if check==0:  
    print "ο αριθμός είναι άρτιος"  
else:  
    print "ο αριθμός είναι περιττός"
```

Ποιο θα είναι το αποτέλεσμα του παραπάνω τμήματος προγράμματος για καθεμία από τις παρακάτω εισόδους;

α. `num=3.5`

β. `num=8`

γ. `num=4.8`

Μονάδες 6

α.

β.

γ.

Έλεγχος

Βαθμολογήθηκε
1,00 στα 1,00
✓
Σημαιοδότηση
ερώτησης

```
num=int(input("Δώστε αριθμό"))
```

```
check=num%2
```

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

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

```
else:
```

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

Ποιο θα είναι το αποτέλεσμα του παραπάνω τμήματος προγράμματος για καθεμία από τις παρακάτω εισόδους;

α. `num=3.5`

β. `num=8`

γ. `num=4.8`

Μονάδες 6

α. Ο αριθμός είναι περιττός. ✓

β. Ο αριθμός είναι άρτιος. ✓

γ. Ο αριθμός είναι άρτιος. ✓

Η `int` μετατρέπει έναν δεκαδικό σε ακέραιο αποκόπτοντας το δεκαδικό μέρος. Δεν κάνει στρογγυλοποίηση.

Ερώτηση 17

Μη πλήρης

Βαθμολογήθηκε
στα 1,00

✓
Σημαιοδότηση
ερώτησης

Αν `a = 1` και `b = 2`, συμπληρώστε τις τιμές που θα έχουν τα `a` και `b` μετά την εκτέλεση των παρακάτω εντολών

```
temp = a
```

```
a = b
```

```
b = temp
```

a

b

Έλεγχος

Ερώτηση **17**

Σωστό

Βαθμολογήθηκε
1,00 στα 1,00🏆
Σημαιοδότηση
ερώτησης

Αν **a = 1** και **b = 2**, συμπληρώστε τις τιμές που θα έχουν τα **a** και **b** μετά την εκτέλεση των παρακάτω εντολών

```
temp = a
```

```
a = b
```

```
b = temp
```

a ✓

b ✓

Αντιμετάθεση των τιμών δυο μεταβλητών με χρήση επιπλέον προσωρινής μεταβλητής.

Μη πλήρης
Βαθμολογήθηκε
στα 1,00
Γ
Σημαιοδότηση
ερώτησης

B1. Ένα φορτηγό εταιρίας μπορεί να μεταφέρει βάρος μέχρι 1.500 κιλά. Η εταιρία το γεμίζει με κιβώτια διαφορετικού βάρους το καθένα. Θεωρούμε ότι το φορτηγό είναι αρχικά άδειο.

Το παρακάτω τμήμα προγράμματος σε γλώσσα Python διαβάζει το βάρος κάθε κιβωτίου σε κιλά μέχρι το συνολικό βάρος των κιβωτίων να μην ξεπεράσει τη δυνατότητα μεταφοράς του φορτηγού. Στη συνέχεια εμφανίζει το πλήθος των κιβωτίων που έχουν τοποθετηθεί στο φορτηγό.

Στο πρόγραμμα υπάρχουν πέντε κενά, τα οποία έχουν αριθμηθεί και υπογραμμιστεί. Να γράψετε στο τετράδιό σας τον αριθμό του κενού και δίπλα τι πρέπει να συμπληρωθεί ώστε το πρόγραμμα να εκτελεί σωστά τη λειτουργία του.

```
size = input("Δώστε βάρος κιβωτίου σε κιλά=")
kivotia = 0                                #πλήθος κιβωτίων
capacity = 1500.0
while (1) <= (2) :
    capacity = (3) - (4)
    kivotia = kivotia + 1
    size = input("Δώστε βάρος κιβωτίου σε κιλά=")
print "Πλήθος κιβωτίων=", (5)
```

Μονάδες 15

(1)

(2) :

(3)

(4)

(5)

B1. Ένα φορτηγό εταιρείας μπορεί να μεταφέρει βάρος μέχρι 1.500 κιλά. Η εταιρεία το γεμίζει με κιβώτια διαφορετικού βάρους το καθένα. Θεωρούμε ότι το φορτηγό είναι αρχικά άδειο.

Το παρακάτω τμήμα προγράμματος σε γλώσσα Python διαβάζει το βάρος κάθε κιβωτίου σε κιλά μέχρι το συνολικό βάρος των κιβωτίων να μην ξεπεράσει τη δυνατότητα μεταφοράς του φορτηγού. Στη συνέχεια εμφανίζει το πλήθος των κιβωτίων που έχουν τοποθετηθεί στο φορτηγό.

Στο πρόγραμμα υπάρχουν πέντε κενά, τα οποία έχουν αριθμηθεί και υπογραμμιστεί. Να γράψετε στο τετράδιό σας τον αριθμό του κενού και δίπλα τι πρέπει να συμπληρωθεί ώστε το πρόγραμμα να εκτελεί σωστά τη λειτουργία του.

```
size = input("Δώστε βάρος κιβωτίου σε κιλά=")
kivotia = 0                                #πλήθος κιβωτίων
capacity = 1500.0
while (1) <= (2) :
    capacity = (3) - (4)
    kivotia = kivotia + 1
    size = input("Δώστε βάρος κιβωτίου σε κιλά=")
print "Πλήθος κιβωτίων=", (5)
```

Μονάδες 15

- (1) ✓
- (2) ✓ :
- (3) ✓
- (4) ✓
- (5) ✓



Αν **a = 1** και **b = 2**, συμπληρώστε τις τιμές που θα έχουν τα **a** και **b** μετά την εκτέλεση των παρακάτω εντολών

a = b

b = a

a

b

Έλεγχος

Ερώτηση 19

Σωστό

Βαθμολογήθηκε
1,00 στα 1,00✓
Σημαιοδότηση
ερώτησης

Αν $a = 1$ και $b = 2$, συμπληρώστε τις τιμές που θα έχουν τα a και b μετά την εκτέλεση των παρακάτω εντολών

 $a = b$ $b = a$ a ✓ b ✓

Καταστρέψαμε την τιμή του a πριν να την μεταφέρουμε στο b .

Αντιμετάθεση των τιμών δυο μεταβλητών **δε** γίνεται έτσι. :)

Ερώτηση **20**

Μη πλήρης

Βαθμολογήθηκε
στα 1,00Σημαιοδότηση
ερώτησης

Για ποιες τιμές του a θα εκτυπωθεί η τιμή 2;

```
a=input()
```

```
if a<0 or a>5:
```

```
    print 1
```

```
elif a>20:
```

```
    print 2
```

```
else:
```

```
    print 3
```

Επιλέξτε ένα:

- ☐ για $0 \leq a \leq 5$
- ☐ ποτέ δεν θα εκτυπωθεί
- ☐ $a > 0$ ή $a < 5$
- ☐ $a < -1000$.
- ☐ για όλες

Έλεγχος

Ερώτηση 20

Σωστό

Βαθμολογήθηκε
1,00 στα 1,00

🏆

Σημαιοδότηση
ερώτησης

Για ποιες τιμές του a θα εκτυπωθεί η τιμή 2;

```
a=input()
```

```
if a<0 or a>5:
```

```
    print 1
```

```
elif a>20:
```

```
    print 2
```

```
else:
```

```
    print 3
```

Επιλέξτε ένα:

- ☐ για $0 \leq a \leq 5$
- ☒ ποτέ δεν θα εκτυπωθεί
- ☐ $a > 0$ ή $a < 5$
- ☐ $a < -1000$.
- ☐ για όλες



Η απαντησή σας είναι σωστή

Για να εκτυπωθεί το 2 πρέπει να δώσουμε μία τιμή μεγαλύτερη από το 20 αλλά τότε θα μπαίναμε στην πρώτη επιλογή καθώς η τιμή αυτή θα ήταν Φυσικά και μεγαλύτερη από το 5

Αυτή δεν είναι μία ωραία χρήση της if, δεν προτείνουμε να χρησιμοποιούμε με τόσο περίεργους τρόπους το if. Απλά αυτές οι ασκήσεις είναι ένας καλός τρόπος να καταλάβουμε πώς λειτουργεί το if.

Οι σωστή απάντηση είναι: ποτέ δεν θα εκτυπωθεί

Προηγούμενη σελίδα

Τέλος προσπάθειας...

Ερώτηση 1

Μη πλήρης

Βαθμολογήθηκε
στα 1,00



Σημαιοδότηση
ερώτησης

Δίνεται το παρακάτω τμήμα προγράμματος Python που υλοποιεί τον Αλγόριθμο Διαδικής αναζήτησης στα στοιχεία της λίστας array η οποία είναι ταξινομημένη **κατά αύξουσα σειρά**. Ο αλγόριθμος επιστρέφει τη θέση του στοιχείου key αν υπάρχει, αλλιώς επιστρέφει -1.

```
def binarySearch( array, key ) :
```

```
    first= 
```

```
    last = 
```

```
    pos = 
```

```
    while first <= last and  :
```

```
        mid = 
```

```
        if array[  ] ==  :
```

```
            pos = mid
```

```
        elif array[ mid ] < key :
```

```
            
```

```
        else
```

```
            
```

```
    return pos
```

0

(first+last)/2

len(array)

last = mid - 1

first = mid - 1

pos== -1

first = mid+1

key

mid

-1

last = mid + 1

len(array)-1

Έλεγχος

Ερώτηση 1

Σωστό

Βαθμολογήθηκε
στα 1,00



Σημαιοδότηση
ερώτησης

Δίνεται το παρακάτω τμήμα προγράμματος Python που υλοποιεί τον Αλγόριθμο Δυναμικής αναζήτησης στα στοιχεία της λίστας array η οποία είναι ταξινομημένη **κατά αύξουσα σειρά**. Ο αλγόριθμος επιστρέφει τη θέση του στοιχείου key αν υπάρχει, αλλιώς επιστρέφει -1.

```
def binarySearch( array, key ) :
```

```
    first=  ✓
```

```
    last =  ✓
```

```
    pos =  ✓
```

```
    while first <= last and  ✓ :
```

```
        mid =  ✓
```

```
        if array[  ✓ ] ==  ✓ :
```

```
            pos = mid
```

```
        elif array[ mid ] < key :
```

```
             ✓
```

```
        else
```

```
             ✓
```

```
    return pos
```

```
def binarySearch( array, key ) :
```

```
    first = 0
```

return pos

Ερώτηση 2

Μη πλήρης

Βαθμολογήθηκε
στα 1,00

📄

Σημαιοδότηση
ερώτησης

B2. Δίνεται παρακάτω η λίστα A.

	0	1	2	3	4	5	6	7	8	9	10	11	12	13
A:	1	1	2	3	5	8	13	21	34	55	89	94	96	99

Να γράψετε στο τετράδιό σας τους αριθμούς της λίστας που θα συγκριθούν κατά την εκτέλεση του αλγορίθμου της δυαδικής αναζήτησης σε κάθε μία από τις παρακάτω περιπτώσεις:

α) για τον αριθμό 100 (μον. 4)

Πανελλήνιες Ιούνιος 2018 Θέμα Β2

Σημείωση : Γράψτε τους αριθμούς που θα συγκριθούν κατά την εκτέλεση του αλγορίθμου στη σειρά χωρισμένους με ένα κενό.

Απάντηση:

Έλεγχος

Ερώτηση 2

Μερικώς σωστή

Βαθμολογήθηκε στα 1,00



Σημαιοδότηση ερώτησης

B2. Δίνεται παρακάτω η λίστα A.

0 1 2 3 4 5 6 7 8 9 10 11 12 13

A:

1	1	2	3	5	8	13	21	34	55	89	94	96	99
---	---	---	---	---	---	----	----	----	----	----	----	----	----

Να γράψετε στο τετράδιό σας τους αριθμούς της λίστας που θα συγκριθούν κατά την εκτέλεση του αλγορίθμου της δυαδικής αναζήτησης σε κάθε μία από τις παρακάτω περιπτώσεις:

α) για τον αριθμό 100 (μον. 4)

Πανελλήνιες Ιούνιος 2018 Θέμα Β2

Σημείωση : Γράψτε τους αριθμούς που θα συγκριθούν κατά την εκτέλεση του αλγορίθμου στη σειρά χωρισμένους με ένα κενό.

Απάντηση: 13 89 96



Η σωστή απάντηση είναι: 13 89 96 99

Ερώτηση **3**

Μη πλήρης
Βαθμολογήθηκε
στα 1,00

🚩
Σημαιοδότηση
ερώτησης

Οι βαθμοί 100 μαθητών έχουν αποθηκευτεί στη λίστα **grade**

```
def bubbleSort5( A):           #φθίνουσα ταξινόμηση
    N = len( A )
    for i in range( 5):        # <====ΠΡΟΣΟΧΗ====
        for j in range(N-1, i, -1):
            if A[ j ] > A[ j-1 ] :
                A[ j ], A[ j-1 ] = A[ j-1 ], A[ j ]
```

Σημείωση: θεωρήστε ότι όλοι οι βαθμοί είναι διαφορετικοί μεταξύ τους
Αντιστοιχίστε το κώδικα με το τι θα εμφανίσει

bubblesort5(grade)

```
for i in range(5):
    print ( grade[i] )
```

bubblesort5(grade)

```
print ( grade[0:5] )
```

Έλεγχος

Επιλέξτε...

Επιλέξτε...

Τους 5 μικρότερους βαθμούς

Τους 5 μεγαλύτερους βαθμούς

Μια λίστα με τους 5 μεγαλύτερους βαθμούς

Μια λίστα με τους 5 μικρότερους βαθμούς

5 τυχαίους βαθμούς

Ερώτηση 3

Μη πλήρης
Βαθμολογήθηκε
στα 1,00

Σημαιοδότηση
ερώτησης

Οι βαθμοί 100 μαθητών έχουν αποθηκευτεί στη λίστα **grade**

```
def bubbleSort5( A):           #φθίνουσα ταξινόμηση
    N = len( A )
    for i in range( 5):        # <====ΠΡΟΣΟΧΗ=====
        for j in range(N-1, i, -1):
            if A[ j ] > A[ j-1 ] :
                A[ j ], A[ j-1 ] = A[ j-1 ], A[ j ]
```

Σημείωση: θεωρήστε ότι όλοι οι βαθμοί είναι διαφορετικοί μεταξύ τους
Αντιστοιχίστε το κώδικα με το τι θα εμφανίσει

bubblesort5(grade)

```
for i in range(5):
    print ( grade[i] )
```

Επιλέξτε...

bubblesort5(grade)

```
print ( grade[0:5] )
```

Επιλέξτε...

Επιλέξτε...

- Τους 5 μικρότερους βαθμούς
- Τους 5 μεγαλύτερους βαθμούς
- Μια λίστα με τους 5 μεγαλύτερους βαθμούς
- Μια λίστα με τους 5 μικρότερους βαθμούς
- 5 τυχαίους βαθμούς

Έλεγχος

Ερώτηση 4

Μη πλήρης

B1. Δίνεται το παρακάτω τμήμα προγράμματος σε γλώσσα προγραμματισμού Python που υλοποιεί το

Στον αλγόριθμο ταξινόμηση φυσαλίδας Bubble sort σε κάθε επανάληψη ένας αριθμός μπαίνει στη θέση του. Αν λοιπόν ψάχνουμε π.χ. **τους τρεις μικρότερους** αριθμούς δεν χρειάζεται να ταξινομήσουμε ολόκληρη τη λίστα των 100 θέσεων αλλά **μόνο τις τρεις πρώτες** στις οποίες θα βρεθούν οι τρεις μικρότεροι αριθμοί.

Αντίστοιχα αν ψάχνουμε **τους τρεις μεγαλύτερους αριθμούς** μπορούμε να κάνουμε **φθίνουσα ταξινόμηση** οπότε **στις τρεις πρώτες θέσεις** μετά από τρεις επαναλήψεις θα βρεθούν οι τρεις μεγαλύτεροι αριθμοί.

Η σωστή απάντηση είναι: bubblesort5(grade)

```
for i in range(5):
```

```
    print ( grade[i] )
```

_____ → Τους 5 μεγαλύτερους βαθμούς,

```
bubblesort5(grade)
```

```
print ( grade[0:5] )
```

_____ → Μια λίστα με τους 5 μεγαλύτερους βαθμούς

Ερώτηση 4

Μη πλήρης

Βαθμολογήθηκε
στα 1,00



Σημαιοδότηση
ερώτησης

B1. Δίνεται το παρακάτω τμήμα προγράμματος σε γλώσσα προγραμματισμού Python που υλοποιεί ταξινόμηση ευθείας ανταλλαγής (bubble sort) στα στοιχεία της λίστας `lista` **κατά φθίνουσα σειρά**:

```
N=len(lista)
```

```
for i in range(  ):
```

```
    for j in range(N-1,  ,  ):
```

```
        if lista[  ]  lista[j-1]:
```

```
            lista[j], lista[j-1] = lista[j-1], lista[j]
```

Στο τμήμα προγράμματος υπάρχουν υπογραμμισμένα κενά τα οποία έχουν αριθμηθεί. Να γράψετε στο τετράδιό σας τους αριθμούς 1, 2, 3, 4 και 5 που αντιστοιχούν στα κενά του παραπάνω τμήματος προγράμματος και δίπλα σε κάθε αριθμό αυτό που πρέπει να συμπληρωθεί ώστε να υλοποιείται σωστά η ταξινόμηση. Μονάδες 10

Έλεγχος

Ερώτηση 4

Σωστό

Βαθμολογήθηκε
στα 1,00



Σημαιοδότηση
ερώτησης

B1. Δίνεται το παρακάτω τμήμα προγράμματος σε γλώσσα προγραμματισμού Python που υλοποιεί ταξινόμηση ευθείας ανταλλαγής (bubble sort) στα στοιχεία της λίστας *lista* **κατά φθίνουσα σειρά**:

```
N=len(lista)

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

Στο τμήμα προγράμματος υπάρχουν υπογραμμισμένα κενά τα οποία έχουν αριθμηθεί. Να γράψετε στο τετράδιό σας τους αριθμούς 1, 2, 3, 4 και 5 που αντιστοιχούν στα κενά του παραπάνω τμήματος προγράμματος και δίπλα σε κάθε αριθμό αυτό που πρέπει να συμπληρωθεί ώστε να υλοποιείται σωστά η ταξινόμηση. Μονάδες 10

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

Η σωστή απάντηση είναι:

B1. Δίνεται το παρακάτω τμήμα προγράμματος σε γλώσσα προγραμματισμού Python που υλοποιεί ταξινόμηση ευθείας ανταλλαγής (bubble sort) στα στοιχεία της λίστας *lista* **κατά φθίνουσα σειρά**:

```
N=len(lista)

for i in range(N-1):
```

Ερώτηση **13**

Μη πλήρης

Βαθμολογήθηκε
στα 1,00



Σημαιοδότηση
ερώτησης

Το όνομα, ο τελικός βαθμός και η πόλη κατοικίας 100 μαθητών αποθηκεύονται σε αντίστοιχες θέσεις στις "παράλληλες" λίστες **name**, **grade**, **city**.

Ο παρακάτω κώδικας:

```
def bubbleSort( A, B, C ):
    N = len( A )
    for i in range( N-1):
        for j in range(N-1, i, -1):
            if A[ j ] > A[ j-1 ]:
                A[ j ], A[ j-1 ] = A[ j-1 ], A[ j ]
                B[ j ], B[ j-1 ] = B[ j-1 ], B[ j ]
                C[ j ], C[ j-1 ] = C[ j-1 ], C[ j ]

bubblesort(grade,name,city)
print ( city[0] )
```

Θα εμφανίσει την πόλη κατοικίας του μαθητή με το μικρότερο βαθμό

Επιλέξτε ένα:

- ☐ Σωστό
- ☐ Λάθος

Έλεγχος

Γίνεται **φθίνουσα** ταξινόμηση ως προς τους βαθμούς οπότε θα εμφανίσει την πόλη κατοικίας του μαθητή με το μεγαλύτερο βαθμό.

ΟΚ, το παραδέχομαι αυτό ήταν ύπουλο.

Η σωστή απάντηση είναι 'False'.

Ερώτηση 14

Μη πλήρης

Βαθμολογήθηκε
στα 1,00



Σημαιοδότηση
ερώτησης

Το όνομα, ο τελικός βαθμός και η πόλη κατοικίας 100 μαθητών αποθηκεύονται σε αντίστοιχες θέσεις στις "παράλληλες" λίστες **name, grade, city**.

Ο παρακάτω κώδικας:

```
def bubbleSort( A, B, C ):
    N = len( A )
    for i in range( N-1):
        for j in range(N-1, i, -1):
            if A[ j ] < A[ j-1 ]:
                A[ j ], A[ j-1 ] = A[ j-1 ], A[ j ]
                B[ j ], B[ j-1 ] = B[ j-1 ], B[ j ]
                C[ j ], C[ j-1 ] = C[ j-1 ], C[ j ]

bubblesort(grade,name,city)
print ( name[0] )
```

Θα εμφανίσει το όνομα του μαθητή με το μικρότερο βαθμό

Γίνεται αύξουσα ταξινόμηση ως προς τους βαθμούς.
Η σωστή απάντηση είναι 'True'.

Ερώτηση **15**

Μη πλήρης

Βαθμολογήθηκε
στα 1,00



Σημαιοδότηση
ερώτησης

Το όνομα, ο τελικός βαθμός και η πόλη κατοικίας 100 μαθητών αποθηκεύονται σε αντίστοιχες θέσεις στις "παράλληλες" λίστες **name**, **grade**, **city**.

Ο παρακάτω κώδικας

```
def bubbleSort( A, B, C):  
    N = len( A )  
    for i in range( N-1):  
        for j in range(N-1, i, -1):  
            if A[ j ] < A[ j-1 ] :  
                A[ j ], A[ j-1 ] = A[ j-1 ], A[ j ]  
                B[ j ], B[ j-1 ] = B[ j-1 ], B[ j ]  
                C[ j ], C[ j-1 ] = C[ j-1 ], C[ j ]  
  
    bubblesort(name,grade,city)  
    print ( grade[-1] )
```

Θα εμφανίσει το μεγαλύτερο βαθμό

Επιλέξτε ένα:

- ☐ Σωστό
☐ Λάθος

Έλεγχος

Γίνεται αύξουσα ταξινόμηση ως προς το όνομα θα εμφανιστεί ο βαθμός του μαθητή με το τελευταίο λεξικογραφικά όνομα .

Η σωστή απάντηση είναι 'False'.

 Αρχική

 Τα μαθήματά μου

 Τρέχον μάθημα

Σ.Ε.Π. Χίου

Erasmus

Ερώτηση 9

Μη πλήρης

Βαθμολογήθηκε
στα 1,00



Σημαιοδότηση
ερώτησης

Το αρχείο κειμένου onomata.txt περιέχει ένα όνομα σε κάθε γραμμή.

Συμπληρώστε το παρακάτω πρόγραμμα ώστε να διαβάζει τα ονόματα από το αρχείο και να τα αποθηκεύει σε μία λίστα.

#Άνοιγμα αρχείου

for line in fin:

List = []

fin = open("onomata.txt","r")

List.append(line)

List = 0

fin = open("onomata.txt","w")

fin = open("onomata.txt","a")

fin.close()

Έλεγχος



Προγραμματισμός Υπολογιστών Python ...

[🏠 Αρχική](#)[📁 Τα μαθήματά μου](#)[👤 Τρέχον μάθημα](#)[Σ.Ε.Π. Χίου](#)[Erasmus](#)[🏠](#) > [Τα μαθήματά μου](#) > [Python Γ ΕΠΑΛ](#) > [Γενική Επανάληψη](#) > [Quiz 6ο Κεφάλαιο Τυχαίες ερωτήσεις 10](#)

Ερώτηση 9

Σωστό

Βαθμολογήθηκε
1,00 στα 1,00



Σημαιοδότηση
ερώτησης

Το αρχείο κειμένου onomata.txt περιέχει ένα όνομα σε κάθε γραμμή.

Συμπληρώστε το παρακάτω πρόγραμμα ώστε να διαβάζει τα ονόματα από το αρχείο και να τα αποθηκεύει σε μία λίστα.

```
fin = open("onomata.txt", "r")
```



#Άνοιγμα αρχείου

```
List = []
```



```
for line in fin:
```



```
List.append(line)
```



```
fin.close()
```



Ερώτηση **10**

Μη πλήρης

Βαθμολογήθηκε
στα 1,00



Σημαιοδότηση
ερώτησης

Συμπληρώστε το παρακάτω πρόγραμμα ώστε να δημιουργηθεί ένα αντίγραφο του αρχείου palio.txt με όνομα neo.txt.

f1 = open("palio.txt","r")

f2.close()

for line in f1:

f2.write(line)

f1.close()

f2 = open("neo.txt","w")

for line in f2:

f1.write(line)

f2 = open("neo.txt","r")

Ερώτηση **10**

Σωστό

Βαθμολογήθηκε
1,00 στα 1,00



Σημαιοδότηση
ερώτησης

Συμπληρώστε το παρακάτω πρόγραμμα ώστε να δημιουργηθεί ένα αντίγραφο του αρχείου palio.txt με όνομα neo.txt.

f1 = open("palio.txt", "r")

f2 = open("neo.txt", "w")



for line in f1:



f2.write(line)



f1.close()



f2.close()

for line in f2:

f1.write(line)

f2 = open("neo.txt", "r")

Η απαντησή σας είναι σωστή

f1 = open("palio.txt", "r")

Ερώτηση 1

Μη πλήρης

Βαθμολογήθηκε
στα 1,00Σημαιοδότηση
ερώτησης

Τι θα εμφανίσει το παρακάτω πρόγραμμα Python:

```
myGlobal = 5  
  
def func1():  
    global myGlobal  
    myGlobal = 42  
    print ( "infunc1 = ", myGlobal )  
  
def func2():  
    print ( "infunc2 = ", myGlobal )  
  
func1()  
func2()
```

infunc1 = infunc2 = **Έλεγχος**

```
myGlobal = 5
def func1():
    global myGlobal
    myGlobal = 42
    print ( "infunc1 = ", myGlobal )
def func2():
    print ( "infunc2 = ", myGlobal )
func1()
func2()
```

infunc1 = 42 ✓

infunc2 = 42 ✓

Δείτε τι έγινε

- Η Python θεωρεί ότι αν μέσα σε μια συνάρτηση εκχωρηθεί σε οποιαδήποτε μεταβλητή μια τιμή, η μεταβλητή αυτή είναι τοπική και η τιμή της ισχύει μόνο για αυτή τη συνάρτηση, **εκτός αν δηλωθεί ρητά διαφορετικά global**.
- Αν απλώς διαβάζει μια τιμή από τη μεταβλητή και αυτή δεν υπάρχει τοπικά, τότε προσπαθεί να την αναζητήσει σε οποιαδήποτε δυνατή εμβέλεια, για παράδειγμα καθολική.
- Αυτή η τοπική μεταβλητή βγαίνει εκτός εμβέλειας και χάνεται, όταν η func1() ολοκληρώνεται.
- Μία συνάρτηση δεν μπορεί να δει τις μεταβλητές που ορίζονται μέσα σε μία άλλη συνάρτηση Εκτός αν αυτές είναι καθολικές.

```
name = "black"
def f():
    print ( "Μέσα στη συνάρτηση", name )

print ( "Έξω από τη συνάρτηση", name )
f()
print ( "Έξω από τη συνάρτηση", name )
```

Επιλέξτε τι θα εμφανίσει το παραπάνω πρόγραμμα:

Έξω από τη συνάρτηση : ✓

Μέσα στη συνάρτηση : ✓

Έξω από τη συνάρτηση : ✓

Μπορούμε να πάρουμε τα περιεχόμενα μιας καθολικής global μεταβλητής μέσα στο σώμα μιας συνάρτησης. Αλλά αν θέλουμε να αλλάξουμε μία καθολική μεταβλητή σε μία συνάρτηση, πρέπει να χρησιμοποιήσουμε τη δήλωση global.

Έξω από τη συνάρτηση: black

Μέσα στη συνάρτηση : black

Έξω από τη συνάρτηση :black



Εξω από τη συνάρτηση: black

```
name = "black"
```

```
def f():
```

```
    name = "white"
```

```
    print ( "Μέσα στη συνάρτηση", name )
```

```
print ( "Εξω από τη συνάρτηση", name )
```

```
f()
```

```
print ( "Εξω από τη συνάρτηση", name )
```

Επιλέξτε τι θα εμφανίσει το παραπάνω πρόγραμμα:

Έξω από τη συνάρτηση:

Μέσα στη συνάρτηση :

Έξω από τη συνάρτηση :

Έλεγχος

```
name = "black"

def f():
    name = "white"
    print ( "Μέσα στη συνάρτηση", name )

print ( "Έξω από τη συνάρτηση", name )
f()
print ( "Έξω από τη συνάρτηση", name )
```

Επιλέξτε τι θα εμφανίσει το παραπάνω πρόγραμμα:

Έξω από τη συνάρτηση: ✓

Μέσα στη συνάρτηση : ✓

Έξω από τη συνάρτηση : ✓

Μία μεταβλητή που ορίζεται στο σώμα της συνάρτησης έχει τοπική εμβέλεια, δηλαδή έχει ισχύ μόνο μέσα στο σώμα της συνάρτησης.

Έξω από τη συνάρτηση: black

Μέσα στη συνάρτηση : white

Έξω από τη συνάρτηση :black



```
def changeme( mylist ):
    mylist.pop()
    print "Τιμές μέσα στη συνάρτηση: ", mylist
```

```
mylist = [1,2,3]
changeme( mylist )
print "Τιμές έξω από τη συνάρτηση: ", mylist
```

Επιλέξτε τι θα εμφανίσει το παραπάνω πρόγραμμα:

Τιμές μέσα στη συνάρτηση :

Τιμές έξω από τη συνάρτηση:

Έλεγχος



```
def changeme( mylist ):
    mylist.pop()
    print "Τιμές μέσα στη συνάρτηση: ", mylist
```

```
mylist = [1,2,3]
changeme( mylist )
print "Τιμές έξω από τη συνάρτηση: ", mylist
```

Επιλέξτε τι θα εμφανίσει το παραπάνω πρόγραμμα:

Τιμές μέσα στη συνάρτηση :

Τιμές έξω από τη συνάρτηση:

Δείτε τι έγινε!

Οι μέθοδοι **insert, append, pop**

και ο τελεστής += **Λίστα += [στοιχείο]**

Προσθέτουν ή αφαιρούν το στοιχείο στην ήδη υπάρχουσα λίστα, οπότε **αλλαγές στη λίστα** μέσα στη συνάρτηση, **επιστρέφουν στα ορίσματα ώστε να γίνουν γνωστές έξω από τη συνάρτηση**.

```
def changeme( a):  
    a[0]=a[0]+1  
    print "Τιμές μέσα στη συνάρτηση: ", a[0]
```

```
x=[1]  
changeme(x )  
print "Τιμές έξω από τη συνάρτηση: ", x[0]
```

Επιλέξτε τι θα εμφανίσει το παραπάνω πρόγραμμα:

Τιμές μέσα στη συνάρτηση :

Τιμές έξω από τη συνάρτηση:

Έλεγχος

Τιμές μέσα στη συνάρτηση: 2

Τιμές έξω από τη συνάρτηση: 2

Τι θα εμφανίσει το παρακάτω πρόγραμμα Python:
Προσοχή! Μόνο ένα κενό ανάμεσα στις τιμές.

```
def mystery(a):  
    k=1  
    return a+k
```

```
x,k = 1,3  
z = mystery(x)  
print z,k
```

Απάντηση:

Έλεγχος

Η σωστή απάντηση είναι: 2 3

Πώς θα προσθέσουμε το "apple" **στο τέλος** της παρακάτω λίστας
fruits = ["watermelon", "banana", "orange"]

(Υπάρχουν πολλές σωστές απαντήσεις!)

Επιλέξτε μία ή παραπάνω σωστές απαντήσεις.

- ☐ fruits.insert(0,"apple")
- ☐ fruits = fruits + ["apple"]
- ☐ append.fruits("apple")
- ☐ fruits.append("apple")
- ☐ fruits += ["apple"]
- ☐ fruits = ["apple"] + fruits
- ☐ fruits.insert(3,"apple")

Εξέρει

Πώς θα προσθέσουμε το "apple" **στο τέλος** της παρακάτω λίστας

```
fruits = ["watermelon", "banana", "orange"]
```

(Υπάρχουν πολλές σωστές απαντήσεις!)

Επιλέξτε μία ή παραπάνω σωστές απαντήσεις.

☐ fruits.insert(0,"apple")

☒ fruits = fruits + ["apple"]

✓ Σωστά

☐ append.fruits("apple")

☒ fruits.append("apple")

✓ Σωστά

☒ fruits += ["apple"]

✓ Σωστά

☐ fruits = ["apple"] + fruits

☒ fruits.insert(3,"apple")

✓ Σωστά

Your answer is correct.

Οι σωστές απαντήσεις είναι: `fruits.append("apple")`, `fruits = fruits + ["apple"]`, `fruits += ["apple"]`, `fruits.insert(3,"apple")`

Λίστες Εφαρμογή. Διαχωρισμός λίστας

Η λειτουργία του διαχωρισμού μιας λίστας σε δύο μέρη με βάση κάποια κριτήρια (θετικοί αρνητικοί), αποτελεί μια τυπική επεξεργασία των λιστών.

```
positives = [ ]
```

```
negatives = [ ]
```

```
for number in numbers :
```

```
    if number > 0:
```

```
         # πρόσθεσέ τον στη λίστα με τους θετικούς
```

```
    else :
```

```
         # αλλιώς στη λίστα με τους αρνητικούς
```

```
print positives
```

```
print negatives
```

Έλεγχος

Λίστες Εφαρμογή. Διαχωρισμός λίστας

Η λειτουργία του διαχωρισμού μιας λίστας σε δύο μέρη με βάση κάποια κριτήρια (θετικοί αρνητικοί), αποτελεί μια τυπική επεξεργασία των λιστών.

```
positives = []
```

```
negatives = []
```

```
for number in numbers :
```

```
    if number > 0:
```

```
        positives = positives + [number]  ✓ # πρόσθεσέ τον στη λίστα με τους θετικούς
```

```
    else :
```

```
        negatives = negatives + [number]  ✓ # αλλιώς στη λίστα με τους αρνητικούς
```

```
print positives
```

```
print negatives
```

Τι θα εμφανιστεί;

```
fruits = ["watermelon", "banana", "orange", "apple"]  
for i in range(0,3,2):  
    print fruits[i]
```

Επιλέξτε ένα:

- ☐ watermelon
banana
orange
- ☐ watermelon
banana
orange
apple
- ☐ watermelon
orange

Έλεγχος

Τι θα εμφανιστεί;

```
fruits = ["watermelon", "banana", "orange", "apple"]  
for i in range(0,3,2):  
    print fruits[i]
```

Επιλέξτε ένα:

- ☐ watermelon
banana
orange
- ☐ watermelon
banana
orange
apple
- ☒ watermelon
orange



Πολύ
σωστά!

Your answer is correct.

Οι σωστή απάντηση είναι: watermelon
orange



Αντιστοιχίστε τον κώδικα με το αποτέλεσμα.

Υποθέστε ότι:

```
singer = "Pink Floyd"
```

```
print ( "singer")
```

```
print ( singer )
```

```
print ( len("singer") )
```

```
print ( len(singer) )
```

```
print ( "Singer:"+singer )
```

```
print ( "Singer: "+"singer" )
```

Επιλέξτε...



Επιλέξτε...

10

6

Pink Floyd

Singer: singer

singer

Singer: Pink Floyd

Μήνυμα λάθους

Η σωστή απάντηση είναι:

```
print ( "singer")→ singer,  
  
print ( singer )→ Pink Floyd,  
  
print ( len("singer") ) → 6,  
  
print ( len(singer) )→ 10,  
  
print ( "Singer:"+singer ) → Singer: Pink Floyd,  
  
print ( "Singer: "+"singer" ) → Singer: singer
```

ΑΚΟΛΟΥΘΕΙ ΑΡΚΕΤΑ ΔΥΣΚΟΛΗ ΑΣΚΗΣΗ



Τα μαθήματά μου



Τρέχον μάθημα

Σ.Ε.Π. Χίου

Erasmus

Θεωρείστε ότι έχουν οριστεί όλες οι απαραίτητες συναρτήσεις ώστε να μπορούμε να χρησιμοποιούμε τη δομή δεδομένων **στοίβα**

```
def push(stack, item)  
def pop(stack)  
def isEmpty(stack)  
def createStack( )
```

Συμπληρώστε τις τιμές που θα εμφανιστούν

```
S=createStack()  
onomata = ["Nikos","Kostas", "Maria"]  
for on in onomata:  
    push(S,on)  
NewS=createStack()  
while not isEmpty(S):  
    push(NewS,pop(S))
```

print pop(NewS)

print pop(NewS)

print pop(NewS)

Nikos
Kostas
Maria

Θεωρείστε ότι έχουν οριστεί όλες οι απαραίτητες συναρτήσεις ώστε να μπορούμε να χρησιμοποιούμε τη δομή δεδομένων **ουρά**

```
def enqueue(queue, item)
def dequeue(queue)
def isEmpty(queue)
def createQueue( )
```

Συμπληρώστε τις τιμές που θα εμφανιστούν

```
oura=createQueue()
onomata = ["Nikos", "Maria"]
for on in onomata:
    enqueue(oura,on)
```

```
print dequeue(oura)
print dequeue(oura)
print dequeue(oura)
```

Έλεγχος

Θεωρείστε ότι έχουν οριστεί όλες οι απαραίτητες συναρτήσεις ώστε να μπορούμε να χρησιμοποιούμε τη δομή δεδομένων **ουρά**

```
def enqueue(queue, item)
```

```
def dequeue(queue)
```

```
def isEmpty(queue)
```

```
def createQueue()
```

Συμπληρώστε τις τιμές που θα εμφανιστούν

```
oura=createQueue()
```

```
onomata = ["Nikos", "Maria"]
```

```
for on in onomata:
```

```
    enqueue(oura,on)
```

```
print dequeue(oura)  
```

```
print dequeue(oura)  
```

```
print dequeue(oura)  
```

Εισάγονται στην ουρά δύο στοιχεία αλλά προσπαθούμε να εξάγουμε τρία, Άρα θα προκληθεί σφάλμα.

Ερώτηση 6

Μη πλήρης

Βαθμολογήθηκε
στα 1,00Σημαιοδότηση
ερώτησης**class Vehicle:****def __init__ (self, color, price, wheels, speed):**

self.color=color

self.price=price

self.wheels=wheels

self.speed=speed

def accelerate(self, amount) :

self.speed += amount

return self.speed

def decelerate(self, amount) :

self.speed -= amount

return self.speed

mybeetle = Vehicle('yellow', 2000.00, 4,80)*Στον παραπάνω κώδικα ποιες μέθοδοι ορίζονται;*

Επιλέξτε μία ή παραπάνω σωστές απαντήσεις.

- ☐ mybeetle
- ☐ price
- ☐ self
- ☐ Vehicle
- ☐ color
- ☐ decelarate
- ☐ speed
- ☐ accelerate
- ☐ __init__
- ☐ wheels

ΠΡΟΣΟΧΗ ΜΕΘΟΔΟΣ ΕΙΝΑΙ ΚΑΙ Ο ΚΑΤΑΣΚΕΥΑΣΤΗΣ!!!!

Οι σωστές απαντήσεις είναι: `accelerate`, `decelarate`, `__init__`

Να υλοποιήσετε την δομή δεδομένων "Στοιβά" ως μια κλάση αξιοποιώντας τις τεχνικές του αντικειμενοστρεφούς προγραμματισμού.

Σημείωση: Η στοιβά σε αυτή την περίπτωση υλοποιείται προσθέτοντας και αφαιρώντας στοιχεία από την αρχή της λίστας.

```
class Stack:
```

```
    def __init__():
```

```
        self.items = []
```

```
    def isEmpty(self):
```

```
        return 
```

```
    def push(self, item):
```

```
        
```

```
    def pop(self):
```

```
        return 
```

Ερώτηση 7

Σωστό

Βαθμολογήθηκε
1,00 στα 1,00



Σημαιοδότηση
ερώτησης

Να υλοποιήσετε την δομή δεδομένων "Στοίβα" ως μια κλάση αξιοποιώντας τις τεχνικές του αντικειμενοστρεφούς προγραμματισμού.

Σημείωση: Η στοίβα σε αυτή την περίπτωση υλοποιείται προσθέτοντας και αφαιρώντας στοιχεία από την αρχή της λίστας.

```
class Stack:
    def __init__(self  ):
        self.items = []
    def isEmpty(self):
        return  
    def push(self, item):
         
    def pop(self):
        return  
```

```
class Stack:
```

```
class Vehicle1:
    def __init__( self, color, price, wheels):
        self.color=color
        self.price=price
        self.wheels=wheels

    def set_speed(self, speed = 100)
        self.speed=speed

    def accelerate(self, amount) :
        self.speed += amount
        return self.speed

    def decelerate(self, amount) :
        self.speed -= amount
        return self.speed

betty=Vehicle1('yellow', 2000.00, 4)
betty.accelerate(10)
```

Θα προκληθεί σφάλμα κατά την εκτέλεση.

Επιλέξτε ένα:

- ☐ Σωστό
- ☐ Λάθος

Έλεγχος

Στη κλάση Vehicle1 αρχικοποιούνται στην __init__ όλες οι ιδιότητες, εκτός από την ταχύτητα (speed) που αρχικοποιείται στην set_speed.

```
betty=Vehicle1('yellow', 2000.00, 4)
```

```
betty.accelerate(10)
```

Λάθος αφού γίνεται απόπειρα να μεταβληθεί η ταχύτητα που δεν έχει αρχικοποιηθεί.

Επομένως, η αρχικοποίηση ιδιοτήτων έξω από την __init__ οδηγεί εύκολα σε λάθη.

Η σωστή απάντηση είναι 'True'.

Ως υπογραφή μεθόδου ορίζεται το όνομα, τα ορίσματα και η τιμή επιστροφής της μεθόδου.

Επιλέξτε ένα:

☐ Σωστό

☐ Λάθος

Έλεγχος

ΣΩΣΤΟ

ΑΚΟΛΟΥΘΕΙ ΔΥΣΚΟΛΗ ΑΣΚΗΣΗ

Δίνεται η παρακάτω κλάση:

```
class Pelatis:
    def __init__( self, onoma, ypoloipo ):
        self.onoma=onoma
        self.ypoloipo = ypoloipo
    def analipsi(self, poso):
        self.ypoloipo -= poso
        return self.ypoloipo
    def katathesi(self, poso):
        self.ypoloipo += poso
        return self.ypoloipo
p1 = Pelatis("Mitsos",2000)
p2 = Pelatis("Maria",3000)
```

Με βάση την παραπάνω ορισμένη κλάση και τα δύο στιγμιότυπα p1,p2 που έχουν ορισθεί:

α) Ορίστε μια μεταβλητή κλάσης με όνομα count η οποία θα μετράει πόσα στιγμιότυπα της κλάσης έχουν ορισθεί.

ΑΠΑΝΤΗΣΗ

```
class Pelatis:
```

```
    no_inst = 0
```

```
    def __init__( self, onoma, ypoloipo ):
```

```
        self.onoma=onoma
```

```
        self.ypoloipo = ypoloipo
```

```
        Pelatis.no_inst = Pelatis.no_inst + 1
```

```
    def analipsi(self, poso):
```

```
        self.ypoloipo -= poso
```

```
        return self.ypoloipo
```

```
    def katathesi(self, poso):
```

```
        self.ypoloipo += poso
```

```
        return self.ypoloipo
```

```
    @classmethod
```

```
    def get_no_of_pelates(cls):
```

```
        return cls.no_inst
```

```
p1 = Pelatis("Mitsos",2000)
```

```
p2 = Pelatis("Maria",3000)
```

```
print Pelatis.get_no_of_pelates()
```

B3. Το παρακάτω τμήμα προγράμματος Python υπολογίζει το γινόμενο $1*2*3*4*5$:

mult= ⁽¹⁾

for i in range (⁽²⁾ **,** ⁽³⁾ **):**

mult=mult* ⁽⁴⁾

print "Το αποτέλεσμα είναι ",mult

Στο τμήμα προγράμματος υπάρχουν τέσσερα (4) κενά, τα οποία έχουν αριθμηθεί και υπογραμμιστεί. Να γράψετε στο τετράδιό σας τον αριθμό του κενού και δίπλα τι πρέπει να συμπληρωθεί, ώστε το τμήμα προγράμματος να εκτελεί σωστά τη λειτουργία του.

Μονάδες 8

(1)

(2)

(3)

(4)

Έλεγχος



Προγραμματισμός Υπολογιστών Python ...

[Αρχική](#)[Τα μαθήματά μου](#)[Τρέχον μάθημα](#)[Σ.Ε.Π. Χίου](#)[Erasmus](#)[🏠](#) > [Τα μαθήματά μου](#) > [Python Γ ΕΠΑΛ](#) > [Γενική Επανάληψη](#) > [Quiz Σύνολο ύλης 20 Τυχαίες ερωτήσεις](#)

Ερώτηση 1

Σωστό

Βαθμολογήθηκε
1,00 στα 1,00



Σημαιοδότηση
ερώτησης

B3. Το παρακάτω τμήμα προγράμματος Python υπολογίζει το γινόμενο $1*2*3*4*5$:

```
mult= (1)_____
```

```
for i in range ( (2)____, (3)____):
```

```
    mult=mult* (4)_____
```

```
print "Το αποτέλεσμα είναι ",mult
```

Στο τμήμα προγράμματος υπάρχουν τέσσερα (4) κενά, τα οποία έχουν αριθμηθεί και υπογραμμιστεί. Να γράψετε στο τετράδιό σας τον αριθμό του κενού και δίπλα τι πρέπει να συμπληρωθεί, ώστε το τμήμα προγράμματος να εκτελεί σωστά τη λειτουργία του.

Μονάδες 8

(1) ✓

(2) ✓

(3) ✓

(4) ✓

(1) 1

Ερώτηση **2**

Μη πλήρης

Βαθμολογήθηκε
στα 1,00



Σημαιοδότηση
ερώτησης

Ο αλγόριθμος ταξινόμησης ευθείας ανταλλαγής (straight exchange sort) είναι ευρύτερα γνωστός ως αλγόριθμος ταξινόμησης φυσαλίδας (bubble sort). Αυτό διότι σε κάθε πέρασμα, το αμέσως μικρότερο στοιχείο ανεβαίνει, όπως μια φυσαλίδα, στην επιφάνεια του νερού.

Επιλέξτε ένα:

☐ Σωστό

☐ Λάθος

Έλεγχος

Η σωστή απάντηση είναι 'True'.

Ερώτηση **3**

Μη πλήρης

Βαθμολογήθηκε
στα 1,00



Σημαιοδότηση
ερώτησης

Λίστες Παράδειγμα 2. Μέσος όρων των στοιχείων μιας λίστας

Για να υπολογίσουμε το μέσο όρο των στοιχείων μιας λίστας, πρώτα χρειάζεται να υπολογίσουμε το άθροισμα των στοιχείων, χρησιμοποιώντας μια μεταβλητή στην οποία προσθέτουμε, κάθε φορά, το επόμενο στοιχείο της λίστας:

sum = 0.0 # το sum είναι πραγματικός (float)

if len(L)>0:

Έλεγχος

Ερώτηση 3

Σωστό

Βαθμολογήθηκε
1,00 στα 1,00Σημαιοδότηση
ερώτησης**Λίστες Παράδειγμα 2.** Μέσος όρων των στοιχείων μιας λίστας

Για να υπολογίσουμε το μέσο όρο των στοιχείων μιας λίστας, πρώτα χρειάζεται να υπολογίσουμε το άθροισμα των στοιχείων, χρησιμοποιώντας μια μεταβλητή στην οποία προσθέτουμε, κάθε φορά, το επόμενο στοιχείο της λίστας:

```
sum = 0.0          # το sum είναι πραγματικός (float)
```

```
for number in L :  ✓
```

```
    sum = sum + number  ✓
```

```
if len(L)>0:
```

```
    average = sum / len( L )  ✓
```

```
    print average  ✓
```

Η απαντησή σας είναι σωστή

Ερώτηση **4**

Μη πλήρης

Βαθμολογήθηκε
στα 1,00



Σημαιοδότηση
ερώτησης

```
name = "black"
```

```
def f():
```

```
    print ( "Μέσα στη συνάρτηση", name )
```

```
print ( "Έξω από τη συνάρτηση", name )
```

```
f()
```

```
print ( "Έξω από τη συνάρτηση", name )
```

Επιλέξτε τι θα εμφανίσει το παραπάνω πρόγραμμα:

Έξω από τη συνάρτηση :

Μέσα στη συνάρτηση :

Έξω από τη συνάρτηση :

Έλεγχος

Ερώτηση 4

Σωστό

Βαθμολογήθηκε
1,00 στα 1,00



Σημαιοδότηση
ερώτησης

```
name = "black"

def f():
    print ( "Μέσα στη συνάρτηση", name )

print ( "Έξω από τη συνάρτηση", name )

f()

print ( "Έξω από τη συνάρτηση", name )
```

Επιλέξτε τι θα εμφανίσει το παραπάνω πρόγραμμα:

Έξω από τη συνάρτηση : ✓

Μέσα στη συνάρτηση : ✓

Έξω από τη συνάρτηση : ✓

Μπορούμε να πάρουμε τα περιεχόμενα μιας καθολικής global μεταβλητής μέσα στο σώμα μιας συνάρτησης. Αλλά αν θέλουμε να αλλάξουμε μία καθολική μεταβλητή σε μία συνάρτηση, πρέπει να χρησιμοποιήσουμε τη δήλωση global.

Έξω από τη συνάρτηση: black

Μέσα στη συνάρτηση : black

Έξω από τη συνάρτηση :black

Ερώτηση 5

Ερώτηση **5**

Μη πλήρης

Βαθμολογήθηκε
στα 1,00



Σημαιοδότηση
ερώτησης

Στην ουρά το στοιχείο που προστέθηκε _____ είναι και το πρώτο που θα εξαχθεί.

Επιλέξτε ένα:

- ☐ τελευταίο
- ☐ πρώτο

Έλεγχος

Οι σωστή απάντηση είναι: πρώτο

Ερώτηση **6**

Μη πλήρης

Βαθμολογήθηκε
στα 1,00



Σημαιοδότηση
ερώτησης

Επιλέξτε την κατάλληλη λογική συνθήκη η οποία να αληθεύει όταν

Ο ακέραιος x είναι πολλαπλάσιο του 4 αλλά όχι του 5

Επιλέξτε μία ή παραπάνω σωστές απαντήσεις.

- ☒ $x \% 4 == 0$ and $x \% 5 == 0$
- ☒ $x \% 4 == 0$ and $x \% 5 != 0$
- ☒ $x \% 4 != 0$ and $x \% 5 != 0$
- ☒ $x \% 4 != 0$ and $x \% 5 == 0$

Έλεγχος

Οι σωστή απάντηση είναι: $x \% 4 == 0$ and $x \% 5 != 0$

Ερώτηση **7**

Μη πλήρης

Βαθμολογήθηκε
στα 1,00



Σημαιοδότηση
ερώτησης

Συμπληρώστε τον πίνακα

P	Q	P or not Q
True	True	
True	False	
False	True	
False	False	

Ερώτηση 7

Σωστό

Βαθμολογήθηκε
1,00 στα 1,00



Σημαιοδότηση
ερώτησης

Συμπληρώστε τον πίνακα

P	Q	P or not Q
True	True	True ✓
True	False	True ✓
False	True	False ✓
False	False	True ✓

```
[isSorted = False]
```

```
[i = i+1]
```

Ερώτηση 9

Μη πλήρης

Βαθμολογήθηκε
στα 1,00Σημαιοδότηση
ερώτησης

Η λίστα, όπως και η συμβολοσειρά, είναι μια δυναμική δομή στην οποία μπορούμε να προσθέτουμε ή να αφαιρούμε στοιχεία (mutable).

Επιλέξτε ένα:

☐ Σωστό☐ Λάθος**Έλεγχος****Ερώτηση 10**

Μη πλήρης

Βαθμολογήθηκε
στα 1,00Σημαιοδότηση
ερώτησης

Ο Αντικειμενοστρεφής προγραμματισμός αλλάζει την εστίαση του προγραμματισμού από τις διαδικασίες στις έννοιες.

Επιλέξτε ένα:

☐ Σωστό☐ Λάθος**Έλεγχος**

Ερώτηση 9

Σωστό

Βαθμολογήθηκε
1,00 στα 1,00

Γ

Σημειοδότηση
ερώτησης

Η λίστα, όπως και η συμβολοσειρά, είναι μια δυναμική δομή στην οποία μπορούμε να προσθέτουμε ή να αφαιρούμε στοιχεία (mutable).

Επιλέξτε ένα:

- ☐ Σωστό
- ☒ Λάθος 

Η λίστα, ~~όπως και η συμβολοσειρά~~, είναι μια δυναμική δομή στην οποία μπορούμε να προσθέτουμε ή να αφαιρούμε στοιχεία (mutable).

Η σωστή απάντηση είναι 'False'.

Ερώτηση 10

Σωστό

Βαθμολογήθηκε
1,00 στα 1,00

Γ

Σημειοδότηση
ερώτησης

Ο Αντικειμενοστρεφής προγραμματισμός αλλάζει την εστίαση του προγραμματισμού από τις διαδικασίες στις έννοιες.

Επιλέξτε ένα:

- ☒ Σωστό 
- ☐ Λάθος

Ο Αντικειμενοστρεφής προγραμματισμός αλλάζει την εστίαση του προγραμματισμού από τις διαδικασίες στις έννοιες.

Σε αυτές αναθέτει **χαρακτηριστικά**, τα οποία ονομάζουμε **ιδιότητες (attributes)**, και τα οποία επεξεργάζεται μέσω ειδικών συναρτήσεων που ονομάζουμε **μεθόδους**.

Οι **ιδιότητες** δεν είναι άλλες από τα σχετικά με το αντικείμενο δεδομένα, και οι **μέθοδοι** αποτελούν **περιγραφές ενεργειών** που αφορούν συγκεκριμένα σε αυτές τις ιδιότητες.

Η σωστή απάντηση είναι 'True'.

Ερώτηση **11**

Μη πλήρης

Βαθμολογήθηκε
στα 1,00



Σημαιοδότηση
ερώτησης

Μια συνάρτηση επιστρέφει μια τιμή με την εντολή `return`.

Επιλέξτε ένα:

☐ Σωστό

☐ Λάθος

Έλεγχος

Ερώτηση **12**

Μη πλήρης

Βαθμολογήθηκε
στα 1,00



Σημαιοδότηση
ερώτησης

1+1

Επιλέξτε μία ή παραπάνω σωστές απαντήσεις.

☐ 2

☐ 10

☐ 11

Έλεγχος

Ερώτηση **13**



Ερώτηση 11

Σωστό

Βαθμολογήθηκε
1,00 στα 1,00



Σημαιοδότηση
ερώτησης

Μια συνάρτηση επιστρέφει μια τιμή με την εντολή `return`.

Επιλέξτε ένα:

- ☒ Σωστό
- ☐ Λάθος

Η σωστή απάντηση είναι 'True'.

Ερώτηση 12

Σωστό

Βαθμολογήθηκε
1,00 στα 1,00



Σημαιοδότηση
ερώτησης

1+1

Επιλέξτε μία ή παραπάνω σωστές απαντήσεις.

- ☒ 2
- ☐ 10
- ☐ 11

OK

Η απαντησή σας είναι σωστή

Οι σωστή απάντηση είναι: 2

Ερώτηση **13**

Σωστό

Βαθμολογήθηκε
1,00 στα 1,00Σημαιοδότηση
ερώτησης

Τι θα εμφανίσει το παρακάτω πρόγραμμα;

```
f= open("monty.txt","w")  
f.write("Always look on the bright side of life")  
f.close()  
f= open("monty.txt","r")  
f.seek(3,0)  
print f.read(2)  
f.close()
```

Επιλέξτε ένα:

- ☐ Θα εμφανιστεί μήνυμα λάθους
- ☒ ay
- ☐ Al
- ☐ wa

Ερώτηση **13**

Σωστό

Βαθμολογήθηκε
1,00 στα 1,00Σημαιοδότηση
ερώτησης

Τι θα εμφανίσει το παρακάτω πρόγραμμα;

```
f= open("monty.txt","w")  
f.write("Always look on the bright side of life")  
f.close()  
f= open("monty.txt","r")  
f.seek(3,0)  
print f.read(2)  
f.close()
```

Επιλέξτε ένα:

- ☐ Θα εμφανιστεί μήνυμα λάθους
- ☒ ay
- ☐ Al
- ☐ wa



Η απαντησή σας είναι σωστή

Ξεκινάμε από την αρχή πάμε 3 θέσεις μετά και διαβάζουμε 2 χαρακτήρες.

Οι σωστή απάντηση είναι: ay

Ερώτηση **14**

Μη πλήρης

Βαθμολογήθηκε
στα 1,00



Σημαιοδότηση
ερώτησης

Τι θα εμφανίσει το παρακάτω;

```
def add(a, b):  
    return a+b  
print add(1,2)
```

Απάντηση:

Έλεγχος

Ερώτηση **15**

Μη πλήρης

Βαθμολογήθηκε
στα 1,00



Σημαιοδότηση
ερώτησης

Τι θα εμφανίσει το παρακάτω;

```
number = 5  
def triple_it(x):  
    print x * 3  
triple_it(number)
```

Απάντηση:

Έλεγχος

Ερώτηση **14**

Σωστό

Βαθμολογήθηκε
1,00 στα 1,00Σημαιοδότηση
ερώτησης

Τι θα εμφανίσει το παρακάτω;

```
def add(a, b):  
    return a+b  
print add(1,2)
```

Απάντηση: 

You got it!

Η σωστή απάντηση είναι: 3

Ερώτηση **15**

Σωστό

Βαθμολογήθηκε
1,00 στα 1,00Σημαιοδότηση
ερώτησης

Τι θα εμφανίσει το παρακάτω;

```
number = 5  
def triple_it(x):  
    print x * 3  
triple_it(number)
```

Απάντηση: 

You got it!

Η σωστή απάντηση είναι: 15

Ερώτηση 16

Μη πλήρης

Βαθμολογήθηκε
στα 1,00Σημαιοδότηση
ερώτησης

Τι θα εμφανίσει ο παρακάτω κώδικας;

```
title = "Chios"  
print title[2]
```

Απάντηση:

Έλεγχος

Ερώτηση 17

Μη πλήρης

Βαθμολογήθηκε
στα 1,00Σημαιοδότηση
ερώτησης

Οι παράμετροι είναι μεταβλητές οι οποίες χρησιμοποιούνται για τη μεταβίβαση τιμών μεταξύ υποπρογραμμάτων ή υποπρογραμμάτων και κύριου προγράμματος.

Επιλέξτε ένα:

☐ Σωστό☐ Λάθος

Έλεγχος

Ερώτηση 18

Μη πλήρης

Βαθμολογήθηκε
στα 1,00Σημαιοδότηση
ερώτησης

2+3*4

Έλεγχος

Ερώτηση 16

Σωστό

Βαθμολογήθηκε
1,00 στα 1,00



Σημαιοδότηση
ερώτησης

Τι θα εμφανίσει ο παρακάτω κώδικας;

```
title = "Chios"
print title[2]
```

Απάντηση: i



Μπράβο!

Η σωστή απάντηση είναι: i

Ερώτηση 17

Σωστό

Βαθμολογήθηκε
1,00 στα 1,00



Σημαιοδότηση
ερώτησης

Οι παράμετροι είναι μεταβλητές οι οποίες χρησιμοποιούνται για τη μεταβίβαση τιμών μεταξύ υποπρογραμμάτων ή υποπρογραμμάτων και κύριου προγράμματος.

Επιλέξτε ένα:

- ☒ Σωστό
- ☐ Λάθος

Οι παράμετροι είναι μεταβλητές οι οποίες χρησιμοποιούνται για τη μεταβίβαση τιμών μεταξύ υποπρογραμμάτων ή υποπρογραμμάτων και κύριου προγράμματος.

7.2 Μεταβλητές και παράμετροι

Όπως έχουμε αναφέρει, ένα υποπρόγραμμα αποτελεί ένα ανεξάρτητο τμήμα προγράμματος που μπορεί να καλείται από σημεία του

7.2 Μεταβλητές και παράμετροι

Όπως έχουμε αναφέρει, ένα υποπρόγραμμα αποτελεί ένα ανεξάρτητο τμήμα προγράμματος που μπορεί να καλείται από σημεία του προγράμματος. Δέχεται τιμές από το τμήμα προγράμματος που το καλεί (το κύριο πρόγραμμα ή άλλο υποπρόγραμμα) και μετά την εκτέλεση των εντολών του επιστρέφει σε αυτό νέες τιμές, δηλαδή τα αποτελέσματα. Οι τιμές αυτές που μεταβιβάζονται από το ένα υποπρόγραμμα στο άλλο λέγονται **παράμετροι** και γενικά σε γλώσσες προγραμματισμού διακρίνονται σε παραμέτρους *Εισόδου* και *Εξόδου*. Οι παράμετροι είναι μεταβλητές οι οποίες χρησιμοποιούνται για τη μεταβίβαση τιμών μεταξύ υποπρογραμμάτων ή υποπρογραμμάτων και κύριου προγράμματος.

Η σωστή απάντηση είναι 'True'.

Ερώτηση 18

Σωστό

Βαθμολογήθηκε
1,00 στα 1,00

Γ
Σημαιοδότηση
ερώτησης

2+3*4

14



Ο πολλαπλασιασμός και η διαίρεση έχουν υψηλότερη προτεραιότητα από την πρόσθεση και αφαίρεση.

2+3*4 =

2+ 12 =

14

Η σωστή απάντηση είναι: 14

Η σωστή απάντηση είναι: 17

Ερώτηση 19

Μη πλήρης

Βαθμολογήθηκε
στα 1,00



Σημαιοδότηση
ερώτησης

Μια συνήθης εφαρμογή του while βρόχου είναι να ελέγχει την εγκυρότητα των δεδομένων εισόδου από το χρήστη

Επιλέξτε ένα:

☐ Σωστό

☐ Λάθος

Έλεγχος

Ερώτηση 20

Μη πλήρης

Βαθμολογήθηκε
στα 1,00



Σημαιοδότηση
ερώτησης

Τα πρότυπα που συγκεντρώνουν τα κοινά στοιχεία άλλων αντικειμένων, έχουν ένα ειδικό όνομα και ονομάζονται κλάσεις (classes). Οι κλάσεις μας επιτρέπουν να γνωρίζουμε πράγματα για αντικείμενα που ακόμα δεν έχουν δημιουργηθεί.

Επιλέξτε ένα:

☐ Σωστό

☐ Λάθος

Έλεγχος

Ερώτηση 19

Σωστό

Βαθμολογήθηκε
1,00 στα 1,00Σημαιοδότηση
ερώτησης

Μια συνηθής εφαρμογή του while βρόχου είναι να ελέγχει την εγκυρότητα των δεδομένων εισόδου από το χρήστη

Επιλέξτε ένα:

- ☒ Σωστό
- ☐ Λάθος

```
num = input("Δώστε ένα θετικό αριθμό")  
while num<0:  
    num = input("Δώστε ένα θετικό αριθμό")  
Η σωστή απάντηση είναι 'True'.
```

Ερώτηση 20

Σωστό

Βαθμολογήθηκε
1,00 στα 1,00Σημαιοδότηση
ερώτησης

Τα πρότυπα που συγκεντρώνουν τα κοινά στοιχεία άλλων αντικειμένων, έχουν ένα ειδικό όνομα και ονομάζονται κλάσεις (classes). Οι κλάσεις μας επιτρέπουν να γνωρίζουμε πράγματα για αντικείμενα που ακόμα δεν έχουν δημιουργηθεί.

Επιλέξτε ένα:

- ☒ Σωστό
- ☐ Λάθος

Κλάση: Ένας τύπος ορισμένος από το χρήστη. Μια συνταγή δημιουργίας αντικειμένων.
Η σωστή απάντηση είναι 'True'.