

**ΠΑΝΕΛΛΑΔΙΚΕΣ ΕΞΕΤΑΣΕΙΣ  
ΗΜΕΡΗΣΙΩΝ ΚΑΙ ΕΣΠΕΡΙΝΩΝ ΕΠΑΓΓΕΛΜΑΤΙΚΩΝ ΛΥΚΕΙΩΝ  
ΤΕΤΑΡΤΗ 20 ΙΟΥΝΙΟΥ 2018  
ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ ΥΠΟΛΟΓΙΣΤΩΝ**

- ΑΠΑΝΤΗΣΕΙΣ -

**ΘΕΜΑ Α**

**A1.**

**α. Σ**

**β. Σ**

**γ. Λ**

**δ. Σ**

**ε. Λ**

**A2.** Δίνεται η παρακάτω κλάση:

```
class Kinito:
    def __init__(self, marka, model):
        self.marka = marka
        self.model = model
    def fortizi(self):
        print "το κινητό φορτίζει"
```

**α)** Ο κατασκευαστής (constructor) της κλάσης είναι η μέθοδος `__init__`

**β)**

```
class Kinito:
    def __init__(self, marka, model, cpu_cores, cam_resolution):
        self.marka = marka
        self.model = model
        self.cpu_cores = cpu_cores
        self.cam_resolution = cam_resolution
    def fortizi(self):
        print "το κινητό φορτίζει"
```

**γ)** `phone1 = Kinito("orange", "S3", 4, 10)`

## ΘΕΜΑ Β

### B1.

Επαναλήψεις		i	j	Εμφανίσεις
for - εξωτερική	for - εσωτερική			
1η	1η	15	2	15 2
	2η		6	15 6
2η	1η	10	2	10 2
	2η		6	10 6

### B2.

α) Για τον αριθμό 100 θα συγκριθούν μεταξύ τους οι αριθμοί :

- στην 1η επανάληψη : 13 με 100
- στην 2η επανάληψη : 89 με 100
- στην 3η επανάληψη : 96 με 100
- στην 4η επανάληψη : 99 με 100

β) Για τον αριθμό 1 θα συγκριθούν μεταξύ τους οι αριθμοί :

- στην 1η επανάληψη : 13 με 1
- στην 2η επανάληψη : 2 με 1
- στην 3η επανάληψη : 1 με 1

### B3.

α)	β)
<pre> i = 20 sum = 0 while i &lt;= 100 :     sum = sum + i     i = i + 20 </pre>	<pre> i = 1 sum = 0 while i &lt;= 5 :     sum = sum + i**2     i = i + 1 </pre>

## ΘΕΜΑ Γ

```
epig1 = raw_input("Δώσε 1η επιγραφή")
```

```
epig2 = raw_input("Δώσε 2η επιγραφή")
```

```
epig = epig1 + epig2
```

```
SUMA = [ ]
```

```
for i in range(24):
```

```
    SUMA = SUMA + [0]
```

```
GRAM=['A', 'B', 'Γ', 'Δ', 'Ε', 'Ζ', 'Η', 'Θ', 'Ι', 'Κ', 'Λ', 'Μ', 'Ν', 'Ξ', 'Ο', 'Π', 'Ρ', 'Σ', 'Τ', 'Υ', 'Φ', 'Χ', 'Ψ', 'Ω'].
```

```
n = len(epig)          # πλήθος γραμμάτων επιγραφής
```

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

```
    key = epig[i]
```

```
    for k in range(24):
```

```
        if GRAM[k] == key:
```

```
            SUMA[k] = SUMA[k] + 1
```

```
pos = -1
```

```
k = 0
```

```
while pos == -1 and i < 24:
```

```
    if GRAM[k] == key:
```

```
        pos = i
```

```
        SUMA[pos] = SUMA[pos] + 1
```

```
    else:
```

```
        k = k + 1
```

```
pl = 0
```

```
for i in range(24):
```

```
    if GRAM[i] != 0:
```

```
        print GRAM[i], 'πρέπει να παραγγελθεί, σε ποσότητα ', SUMA[i]
```

```
    else:
```

```
        pl = pl + 1
```

```
print 'Πλήθος γραμμάτων που δεν θα παραγγελθούν: ', pl
```

## **ΘΕΜΑ Δ**

```
POL = [ ]
THER = [ ]
```

```
f1 = open ('pth.txt', 'r')
```

```
for line in f1 :
    if line % 2 == 1 :
        POL.append(line)
    else :
        THER.append(line)
```

```
s = 0.0
```

```
for i in range(n):
    s = s + THER[i]
```

```
MO = s / n
```

```
print "Μέσος όρος : ", MO
```

```
for i in range(1, n, 1) :
    for j in range(n-1, i-1, -1) :
        if THER[j] > THER[j-1] :
            THER[j], THER[j-1] = THER[j-1], THER[j]
            POL[j], POL[j-1] = POL[j-1], POL[j]
```

```
for i in range(n):
    if THER[i] == THER[0] :           # το THER[0] είναι το μέγιστο , αφού ταξινομήθηκε
        print POL[i]
```

---

### **Σημείωση :**

Τα προγράμματα των Θεμάτων Γ και Δ είναι γραμμένα στην Python 2.7.12.

Για να εμφανιστούν τα μηνύματα με ελληνικά γράμματα, πρέπει να προσθέσουμε στην αρχή του κάθε προγράμματος την γραμμή

```
# -*- coding: utf-8 -*-
```

Αν τα μηνύματα είναι γραμμένα με λατινούς χαρακτήρες, δεν χρειάζεται η παραπάνω γραμμή.

---

Επιμέλεια : Άρης Κεσογλίδης