

ΠΑΝΕΛΛΗΝΙΕΣ - ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ ΥΠΟΛΟΓΙΣΤΩΝ 2017
ΕΠΕΞΗΓΗΣΗ ΤΩΝ ΘΕΜΑΤΩΝ ΘΕΩΡΙΑΣ A4 & B1

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:

Μονάδες 6

x	Πλήθος συμβόλων "**"
2	
3	
4	

Επεξήγηση

1η περίπτωση --> x=2 Ο κώδικας παίρνει την πιο κάτω μορφή:

```
for i in range(1,2):  
    for j in range(1,3):  
        print "**"
```

range(1,2) --> [1]

Η συνάρτηση **range(1,2)** παράγει τη λίστα [1]. Αυτή είναι η λίστα με τις τιμές που πρέπει να διατρέξει το i. Μέσα στη λίστα που παράγει η range () βρίσκεται **μία μόνο τιμή** άρα θα έχω **μία μόνο επανάληψη**. Έτσι η εντολή

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

range(1,3) --> [1,2]

που βρίσκεται στο εσωτερικό της for i ... θα εκτελεστεί **μία μόνο φορά** και θα τυπώσει στην οθόνη 2 αστεράκια.

	i		j	Εμφάνιση στην οθόνη
Επανάληψη 1	1			
		Επανάληψη 1	1	*
		Επανάληψη 2	2	*

2η περίπτωση --> x=3 Ο κώδικας παίρνει την πιο κάτω μορφή:

```
for i in range(1,3):           range(1,3) --> [1,2]
    for j in range(1,3):
        print "*"

```

Η συνάρτηση **range(1,3)** παράγει τη λίστα **[1,2]**. Αυτή είναι η λίστα με τις τιμές που πρέπει να διατρέξει το i. Μέσα στη λίστα που παράγει η range () βρίσκονται **δύο τιμές** άρα θα έχω **δύο επαναλήψεις**. Έτσι η εντολή

```
for j in range(1,3):           range(1,3) --> [1,2]
    print "*"

```

που βρίσκεται στο εσωτερικό της for i ... θα εκτελεστεί **δύο φορές** και θα τυπώσει στην οθόνη **συνολικά 4** αστεράκια.

	i		j	Εμφάνιση στην οθόνη
Επανάληψη 1	1			
		Επανάληψη 1	1	*
		Επανάληψη 2	2	*
Επανάληψη 2	2			
		Επανάληψη 1	1	*
		Επανάληψη 2	2	*

3η περίπτωση --> x=3 Ο κώδικας παίρνει την πιο κάτω μορφή:

```
for i in range(1,4):           range(1,4) --> [1,2,3]
    for j in range(1,3):
        print "*"

```

Η συνάρτηση **range(1,4)** παράγει τη λίστα **[1,2,3]**. Αυτή είναι η λίστα με τις τιμές που πρέπει να διατρέξει το i. Μέσα στη λίστα που παράγει η range () βρίσκονται **τρεις τιμές** άρα θα έχω **τρεις επαναλήψεις**. Έτσι η εντολή

```
for j in range(1,3):           range(1,3) --> [1,2]
    print "*"

```

που βρίσκεται στο εσωτερικό της for i ... θα εκτελεστεί τώρα **τρεις φορές** και θα τυπώσει στην οθόνη **συνολικά 6** αστεράκια.

	i		j	Εμφάνιση στην οθόνη
Επανάληψη 1	1			
		Επανάληψη 1	1	*
		Επανάληψη 2	2	*
Επανάληψη 2	2			
		Επανάληψη 1	1	*
		Επανάληψη 2	2	*
Επανάληψη 3	3			
		Επανάληψη 1	1	*
		Επανάληψη 2	2	*

Απάντηση

x	Πλήθος συμβόλων "*"
2	2
3	4
4	6

B1. Δίνεται το παρακάτω τμήμα προγράμματος σε γλώσσα προγραμματισμού Python:

```
suma=0
a=2
for i in range(10, 14):
    if i%2==1:
        a=a*2
    else:
        suma=suma+a
```

Επίσης δίνεται το παρακάτω υπόδειγμα-πίνακας (πίνακας τιμών):

	i	suma	a
Αρχικές Τιμές		0	2
1η επανάληψη			
...	...	...	...

Να μεταφέρετε στο τετράδιό σας τον παραπάνω πίνακα και να τον συμπληρώσετε, προσθέτοντας μία νέα γραμμή στον πίνακα για κάθε επανάληψη του προγράμματος.

Μονάδες 12

Επεξήγηση

Το πρόγραμμα ξεκινά με τις εντολές

```
suma=0
```

```
a=2
```

Ο υπολογιστής αρχικοποιεί δύο μεταβλητές τις **suma** και **a** στις οποίες δίνει τις τιμές **0** και **2** αντίστοιχα.

Αυτό αντικατοπτρίζεται στην **1η γραμμή** του πιο κάτω πίνακα που έχει στα αριστερά της τη φράση **Αρχικές Τιμές**. Στην ίδια γραμμή και στη στήλη **i** το σχετικό κελί είναι γκριζαρισμένο γιατί το **i** δεν έχει πάρει ακόμη τιμή.

	i	suma	a
Αρχικές Τιμές		0	2

Στη συνέχεια ξεκινά η εντολή επανάληψης **for**

```
for i in range(10, 14):           range(10,14) ---> [10,11,12,13]
    if i%2==1:
        a=a*2
    else:
        suma=suma+a
```

Η **range(10,14)** παράγει τη λίστα **[10,11,12,13]** αποτελούμενη από 4 στοιχεία. Αυτό σημαίνει ότι θα έχω 4 επαναλήψεις. Δηλ. η εντολή **if** που βρίσκεται στο εσωτερικό της **for** θα εκτελεστεί 4 φορές. Προσθέτω λοιπόν στον πιο πάνω πίνακα άλλες 4 γραμμές όσες οι επαναλήψεις ως εξής:

	i	suma	a
Αρχικές Τιμές		0	2
1η επανάληψη			
2η επανάληψη			
3η επανάληψη			
4η επανάληψη			

Στην πρώτη επανάληψη το **i** θα πάρει τη τιμή 10 απ' την πιο πάνω λίστα το συμπληρώνω στη σχετική γραμμή.

Στην δεύτερη επανάληψη το i θα πάρει τη τιμή 11 απ' την πιο πάνω λίστα το συμπληρώνω στη σχετική γραμμή. κοκ.

	i	suma	a
Αρχικές Τιμές		0	2
1η επανάληψη	10		
2η επανάληψη	11		
3η επανάληψη	12		
4η επανάληψη	13		

1η επανάληψη

Όταν το i έχει τιμή 10 τότε ο υπολογιστής θα εκτελέσει για πρώτη φορά την εντολή if. Αυτή θα πάρει την πιο κάτω μορφή :

if i%2==1 :

a=a*2

else:

suma=suma+a

-->

if 10%2==1 :

a=a*2

else :

suma=suma+a

Το υπόλοιπο της διαίρεσης του 10 με το 2 (10%2) είναι 0 άρα η συνθήκη είναι False ο έλεγχος μεταφέρεται στο else και εκτελείται η εντολή suma=suma+a η οποία αυξάνει κατά 2 το περιεχόμενο της suma. Αυτό ήταν 0 και γίνεται τώρα 2. Συμπληρώνω την σχετική γραμμή στον πίνακα. Το a δεν αλλάζει παραμένει 2.

	i	suma	a
Αρχικές Τιμές		0	2
1η επανάληψη	10	2	2
2η επανάληψη	11		
3η επανάληψη	12		
4η επανάληψη	13		

2η επανάληψη

Όταν το i πάρει τιμή 11 τότε ο υπολογιστής θα εκτελέσει για **2η** φορά την εντολή `if`. Αυτή θα πάρει την πιο κάτω μορφή :

`if i%2==1 :`

`a=a*2`

`else:`

`suma=suma+a`

`-->`

`if 11%2==1 :`

`a=a*2`

`else :`

`suma=suma+a`

Το υπόλοιπο της διαίρεσης του 11 με το 2 ($11\%2$) είναι 1 άρα η συνθήκη είναι **True** και έτσι εκτελείται η εντολή `a=a*2` η οποία πολ/ζει με το 2 το τρέχον περιεχόμενο της a (το 2). Κάνει λοιπόν την πράξη $2*2$ και αποθηκεύει το 4 σαν νέα τιμή της a . Συμπληρώνω την σχετική γραμμή στον πίνακα. Το $suma$ δεν αλλάζει παραμένει 2.

	i	$suma$	a
Αρχικές Τιμές		0	2
1η επανάληψη	10	2	2
2η επανάληψη	11	2	4
3η επανάληψη	12		
4η επανάληψη	13		

3η επανάληψη

Όταν το i πάρει τιμή 12 τότε ο υπολογιστής θα εκτελέσει για **3η** φορά την εντολή `if`. Αυτή θα πάρει την πιο κάτω μορφή :

`if i%2==1 :`

`a=a*2`

`else:`

`suma=suma+a`

`-->`

`if 12%2==1 :`

`a=a*2`

`else :`

`suma=suma+a`

Το υπόλοιπο της διαίρεσης του 12 με το 2 ($12\%2$) είναι 0 άρα η συνθήκη είναι **False** ο έλεγχος μεταφέρεται στο `else` και εκτελείται η εντολή `suma=suma+a` η οποία αυξάνει κατά 4 το περιεχόμενο της $suma$. Αυτό ήταν 2 και γίνεται τώρα 6. Συμπληρώνω την σχετική γραμμή στον πίνακα. Το a δεν αλλάζει παραμένει 4.

	i	suma	a
Αρχικές Τιμές		0	2
1η επανάληψη	10	2	2
2η επανάληψη	11	2	4
3η επανάληψη	12	6	4
4η επανάληψη	13		

4η επανάληψη

Όταν το i πάρει τιμή 13 τότε ο υπολογιστής θα εκτελέσει για **4η** φορά την εντολή if. Αυτή θα πάρει την πιο κάτω μορφή :

if $i \% 2 == 1$:

a=a*2

else:

suma=suma+a

-->

if $13 \% 2 == 1$:

a=a*2

else :

suma=suma+a

Το υπόλοιπο της διαίρεσης του 13 με το 2 ($13 \% 2$) είναι 1 άρα η συνθήκη είναι **True** και έτσι εκτελείται η εντολή $a=a*2$ η οποία πολ/ζει με το 2 το τρέχον περιεχόμενο της a (το 2). Κάνει λοιπόν την πράξη $4*2$ και αποθηκεύει το 8 σαν νέα τιμή της a. Συμπληρώνω την σχετική γραμμή στον πίνακα. Το suma δεν αλλάζει παραμένει 6.

	i	suma	a
Αρχικές Τιμές		0	2
1η επανάληψη	10	2	2
2η επανάληψη	11	2	4
3η επανάληψη	12	6	4
4η επανάληψη	13	6	8

Απάντηση

	i	suma	a
Αρχικές Τιμές		0	2
1η επανάληψη	10	2	2
2η επανάληψη	11	2	4
3η επανάληψη	12	6	4
4η επανάληψη	13	6	8