

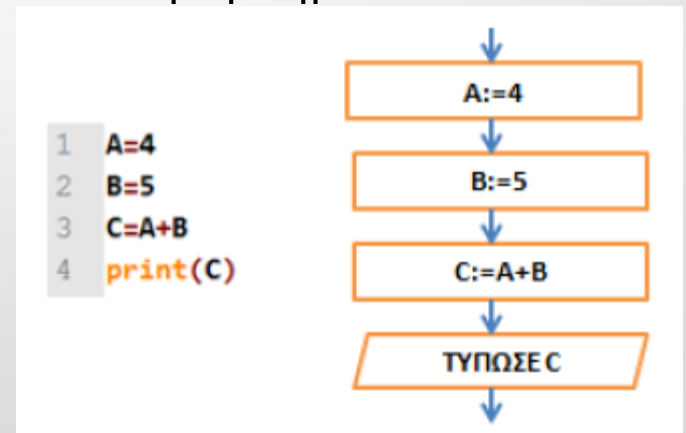
Python: Δομές Ελέγχου Ροής

Ενότητες

- Δομές Ελέγχου Ροής
- Δομή της Απόφασης
 - Η δομή if – elif - else
- Δομές της Επανάληψης
 - Η δομή for
 - Η δομή while

Δομές Ελέγχου Ροής

- Τα προγράμματα που γράψαμε μέχρι τώρα αποτελούνταν από μια σειρά από εντολές που εκτελούνταν από την Python πιστά η μία μετά την άλλη
- Η ακολουθιακή (σειριακή) αυτή δομή εντολών χρησιμοποιείται πρακτικά για την αντιμετώπιση πολύ απλών προβλημάτων.
- Σε πιο σύνθετα προβλήματα απαιτείται η χρήση κατάλληλων εντολών για έλεγχο συνθηκών



Δομές Ελέγχου Ροής

- Οι δυο βασικές δομές ελέγχου ροής προγράμματος είναι:
 - η δομή της απόφασης (**if**)
 - η δομή της επανάληψης (**for, while**)

Δομή της απόφασης

- Στη δομή της απόφασης έχουμε μία πολύπλοκη δομή, στην οποία ο έλεγχος του προγράμματος καλείται να επιλέξει ανάμεσα σε δύο ή και περισσότερες διαφορετικές διαδρομές ανάλογα με το αν ισχύει ή όχι κάποια ή κάποιες συνθήκες
- Η δομή If – elif -else

Η δομή If – elif -else

- Η λογική απόφαση υλοποιείται με τη δομή ελέγχου **if-elif-else**

if condition:

statement_1

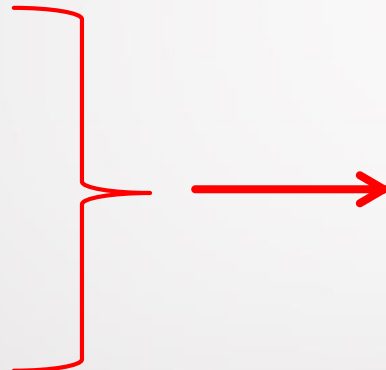
statement_2

...

statement_N

statement_N+1

statement_N+2



Ομαδοποίηση εντολών που ανήκουν στην if με χρήση της εσοχής

Δεν υπάρχει άλλος τρόπος ομαδοποίησής τους, όπως άγκιστρα ή λέξεις κλειδιά που χρησιμοποιούν οι άλλες γλώσσες

Η δομή If – elif -else

- Δηλαδή, ελέγχεται μία συνθήκη και ανάλογα με το αποτέλεσμα (Αληθής ή Ψευδής) εκτελείται ή δεν εκτελείται μία ή κάποια άλλη ομάδα (μπλοκ) εντολών
- Η εντολή if – elif - else συντάσσεται ως εξής:

```
if συνθήκη:  
    μπλοκ_εντολών_1  
else:  
    μπλοκ_εντολών_2
```

```
if συνθήκη:  
    μπλοκ_εντολών_1  
elif συνθήκη:  
    μπλοκ_εντολών_2  
else:  
    μπλοκ_εντολών_3
```

Η δομή If – elif -else

- Παράδειγμα 1

```
x = int(input('Δώστε έναν ακέραιο αριθμό:'))
if x % 2 == 0:
    print(x, 'είναι άρτιος')
else:
    print(x, 'είναι περιττός')
```

```
>>> ===== RESTART ===
>>>
Δώστε έναν ακέραιο αριθμό:4
4 είναι άρτιος
```

```
>>> ===== RESTART ===
>>>
Δώστε έναν ακέραιο αριθμό:7
7 είναι περιττός
```


Η δομή If – elif -else

- Παράδειγμα 2

```
x = int(input('Δώστε έναν ακέραιο αριθμό:'))  
if x > 0:  
    print(x, 'είναι θετικός αριθμός')  
print('Τέλος προγράμματος')
```

```
>>> ===== RESTART ===  
>>>  
Δώστε έναν ακέραιο αριθμό:10  
10 είναι θετικός αριθμός  
Τέλος προγράμματος
```

```
>>> ===== RESTART ===  
>>>  
Δώστε έναν ακέραιο αριθμό:-5  
Τέλος προγράμματος
```

Η δομή If – elif -else

- Παράδειγμα 3

```
x = 20
y = 10

if x < y:
    print(x, '<', y)
elif x > y:
    print(x, '>', y)
else:
    print(x, '=', y)
```

```
>>> ===== RESTART =====
>>>
20 > 10
```

Η δομή If – elif -else

- Παράδειγμα 4 (εμφωλευμένες συνθήκες)

```
x = 20
y = 10

if x == y:
    print(x, '=', y)
else:
    if x < y:
        print(x, '<', y)
    else:
        print(x, '>', y)
```

```
>>> ===== RESTART ===
>>>
20 > 10
```

Η δομή If – elif -else

- Άσκηση: Γράψτε ένα πρόγραμμα που θα ενημερώνει το χρήστη αν ο αριθμός που εισήγαγε είναι θετικός ή αρνητικός. Λάβετε υπόψη και την περίπτωση του μηδέν.

Η δομή If – elif -else

- Άσκηση: επίλυση

```
1 x=int(input('Δώσε μου έναν αριθμό: '))
2 if (x>0):
3     print ('Ο αριθμός είναι θετικός')
4 if (x<0):
5     print ('Ο αριθμός είναι αρνητικός')
6 if (x==0):
7     print ('Ο αριθμός δεν είναι θετικός, ούτε αρνητικός')
```

- Υλοποιήστε το ίδιο με χρήση της If – elif -else

Δομές της επανάληψης

- Θα μπορούσαμε να πούμε ότι οι δύο πιο χαρακτηριστικές δομές επανάληψης είναι:
 - η δομή **for**
 - η δομή **while**

Η δομή for

- Η εντολή for επαναλαμβάνει ένα δεδομένο μπλοκ εντολών για συγκεκριμένο πλήθος φορών

for i in range([A,]B,step)

Statements

όπου *i* η μεταβλητή της επανάληψης, **A** η αρχική τιμή της (οι αγκύλες σημαίνουν ότι είναι προαιρετική, αν δεν δηλωθεί θεωρείται ότι είναι μηδέν), **B** η τελική τιμή της χωρίς όμως να πάρει την τιμή αυτή, **step** το βήμα και **statements** οι εντολές που αποτελούν το σώμα της επανάληψης, οι οποίες είναι και στοιχισμένες μεταξύ τους και λίγο πιο δεξιά από τη στοίχιση της **for**

Η δομή for

- Παράδειγμα 1

```
for i in range(5, 10):  
    print(i)
```

```
>>> ===== RESTART =====  
>>>  
5  
6  
7  
8  
9
```


Η δομή for

- Παράδειγμα 2

```
for i in range(1, 20, 5):  
    print(i)
```

```
>>> ===== RESTART ===  
>>>  
1  
6  
11  
16
```

Η δομή for

- Παράδειγμα 3

```
>>> for i in range(100, 1, -10):  
    print(i)  
  
100  
90  
80  
70  
60  
50  
40  
30  
20  
10  
>>>
```

Η δομή for

- Παράδειγμα 4

```
1 for i in range(1,11):  
2     print(i)  
3  
4 for i in range(1,11,2):  
5     print(i)  
6  
7 for i in range(0,5):  
8     for j in range (0,5):  
9         print (i,j)
```



Η δομή while

- Η εντολή while επαναλαμβανόμενα εκτελεί μία ομάδα (μπλοκ) εντολών, όσο μια συνθήκη (προϋπόθεση) παραμένει αληθής

while (condition):

statements

Η δομή while

- **condition** είναι μία συνθήκη
- **statements** είναι μία ακολουθία εντολών
- Όταν ο έλεγχος του προγράμματος εισέρχεται στη **while**, τότε ελέγχεται εάν η συνθήκη **condition** ισχύει ή όχι. Εάν η συνθήκη δεν ισχύει, τότε ο έλεγχος φεύγει από τη **while**. Αν ισχύει, τότε εκτελούνται οι εντολές **statements** και μετά ο έλεγχος μεταφέρεται στην αρχή της δομής **while**, όπου και ξαναελέγχεται η συνθήκη. Οι **statements** δηλαδή θα εκτελούνται για όσο καιρό η συνθήκη **condition** ισχύει

Η δομή while

- Παράδειγμα 1

```
i = 0
while i < 10:
    print(i)
    i = i + 1
```

```
>>> ===== RESTART =====
>>>
0
1
2
3
4
5
6
7
8
9
```

Η δομή while

- Παράδειγμα 2

```
1 i=1
2 while i<11:
3     print(i)
4     i+=1
5
6 i=1
7 while i<11:
8     print(i)
9     i+=2
10
11 i=1
12 while i<5:
13     j=1
14     while j<5:
15         print (i,j)
16         j+=1
17     i+=1
```



Η δομή while

- Η εντολή **break** χρησιμοποιείται για άμεση έξοδο από οποιοδήποτε σημείο ενός βρόγχου

```
sum = 0
while True:
    n = int(input('Δώστε έναν θετικό αριθμό (-1 για τέλος):'))
    if n == -1:
        break
    sum = sum + n
print('Άθροισμα =', sum)
```

```
>>> ===== RESTART =====
>>>
Δώστε έναν θετικό αριθμό (-1 για τέλος):4
Δώστε έναν θετικό αριθμό (-1 για τέλος):3
Δώστε έναν θετικό αριθμό (-1 για τέλος):10
Δώστε έναν θετικό αριθμό (-1 για τέλος):3
Δώστε έναν θετικό αριθμό (-1 για τέλος):-1
Άθροισμα = 20
```


Ασκήσεις

1. Γράψτε κώδικα ο οποίος να περιέχει έναν βρόγχο for για αντίστροφη εκτύπωση των αριθμών από το -100 έως και το 100 ανά 2.
2. Γράψτε κώδικα ο οποίος να διαβάζει την ηλικία σας (A) και στη συνέχεια να εκτυπώνει αντίστροφα τους αριθμούς από το -A έως και το A. Τέλος, να τυπώνει τους άρτιους αριθμούς από το 1 μέχρι να φτάσει στην ηλικία σας (A).

Βιβλιογραφία

- Μανής, Γ., 2015. Εισαγωγή στον Προγραμματισμό με αρωγό τη γλώσσα Python. [ηλεκτρ. βιβλ.] Αθήνα: Σύνδεσμος Ελληνικών Ακαδημαϊκών Βιβλιοθηκών. Διαθέσιμο στο: <http://hdl.handle.net/11419/2745>
- Αγγελιδάκης, Ν., 2015. Εισαγωγή στον προγραμματισμό με την Python, Ηράκλειο. [ηλεκτρ. βιβλ.] Διαθέσιμο στο: http://aggelid.mysch.gr/pythonbook/INTRODUCTION_TO_COMPUTER_PROGRAMMING_WITH_PYTHON.pdf