



Προγραμματισμός Υπολογιστών

Δομές επανάληψης (1-3)

Νικόλαος Ζ. Ζάχαρης
Καθηγητής

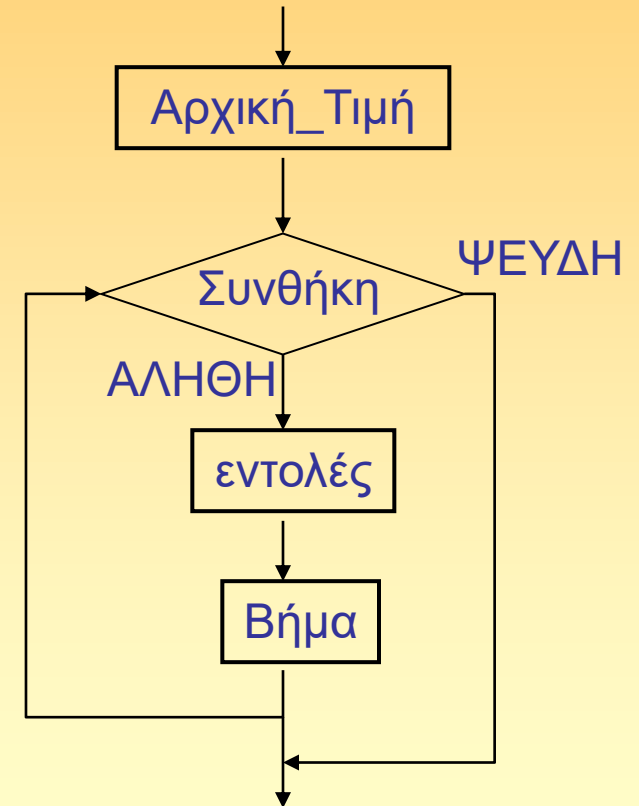
Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Η δομή επανάληψης for

Οι δηλώσεις if και switch μας επιτρέπουν την εκτέλεση εντολών στην περίπτωση που αληθεύει μία συνθήκη. Η δήλωση for μας επιτρέπει την επανάληψη των εντολών ενώ αληθεύει μία συνθήκη. Η σύνταξη της for είναι όπως παρακάτω :

```
for(Αρχική_Τιμή; Συνθήκη; Βήμα) {  
    εντολές;  
}
```

Όπως φαίνεται και στο διάγραμμα ροής η for εκτελείται σε διακριτά στάδια. Αρχικά εκτελείται, για μια μόνο φορά, η δήλωση **Αρχική_Τιμή**. Εν συνεχεία γίνεται εκτίμηση της **Συνθήκης** το αποτέλεσμα της οποίας αν είναι 0 (ΨΕΥΔΗΣ) οδηγεί στον τερματισμό της επανάληψης for, ενώ αν έχει τιμή διάφορη του 0 (ΑΛΗΘΗΣ) τότε εκτελούνται οι **εντολές**. Τέλος εκτελείται η δήλωση **Βήμα** και η ροή του προγράμματος συνεχίζεται με την εκτίμηση της **Συνθήκης**.



Η δομή επανάληψης for

Ο πιο εύκολος τρόπος για τη δημιουργία μιας επανάληψης for είναι με τη βοήθεια ενός ακέραιου μετρητή, όπου αρχικά του αναθέτουμε μία τιμή (**Αρχική_Τιμή**), εν συνέχεια γίνεται μέσω μιας συνθήκης, η σύγκριση της τιμής του μετρητή με ένα όριο (**Συνθήκη**), και τέλος προσδιορίζουμε τη μεταβολή της τιμής του μετρητή μετά το πέρας της εκτέλεσης των εντολών της επανάληψης (**Βήμα**), η οποία μπορεί να γίνει είτε κατά ένα θετικό αριθμό ($i = i + 4$) είτε κατά αρνητικό αριθμό ($i = i - 4$).

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
int main(int argc, char *argv[]) {
```

```
    int i;
```

```
    for (i = 1; i < 4; i++)
```

```
    {
```

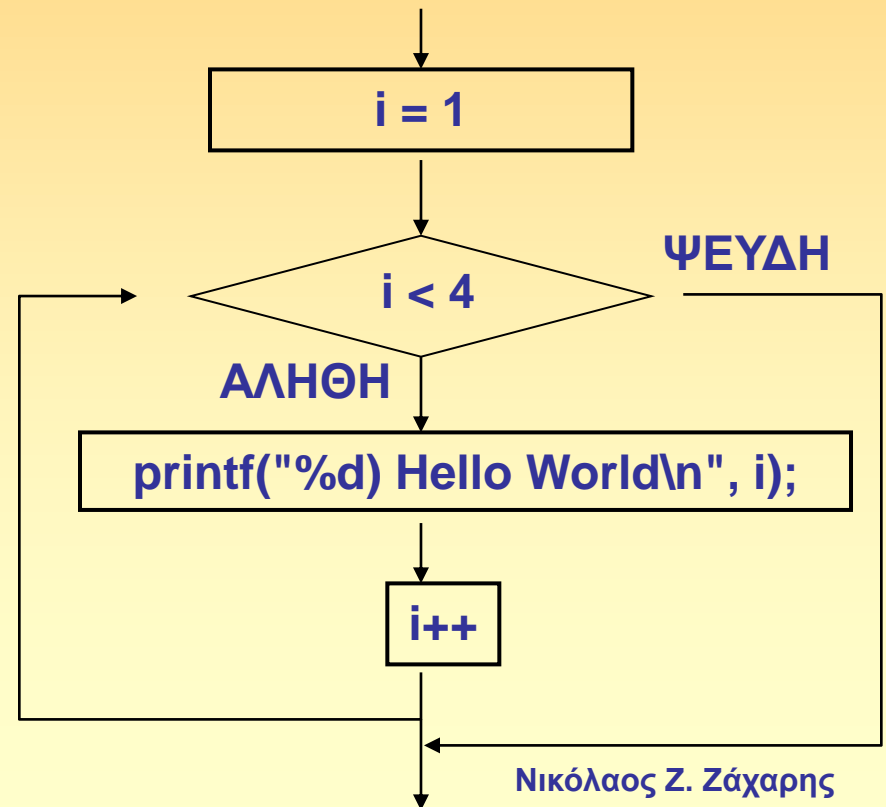
```
        printf("%d) Hello World\n", i);
```

```
    }
```

```
    return 0;
```

```
}
```

```
1) Hello World
2) Hello World
3) Hello World
```



Παράδειγμα

Να διαβάσετε ένα ακέραιο αριθμό, έστω A , από το πληκτρολόγιο. Να γίνεται έλεγχος αν είναι μεγαλύτερος από το 0. Αν δεν είναι να εμφανίζεται το μήνυμα "Invalid Number", διαφορετικά να εμφανίζετε το άθροισμα όλων των αριθμών από το 1 έως τον A .

Αρχή Αλγόριθμου

Διάβασε A

Αν ($A < 0$) Τότε

Εκτύπωσε "Invalid Number"

Διαφορετικά

$sum = 0$

Για $i = 0$ ΜΕΧΡΙ A Με Βήμα 1

$sum = sum + i$

Τέλος Επανάληψης

Εκτύπωσε sum

Τέλος Αν

Τέλος Αλγόριθμου

```
#include <stdio.h>
#include <stdlib.h>
```

```
int main(int argc, char *argv[]) {
    int i, A;
    long int sum = 0;
    printf("Type an integer number : ");
    scanf("%d", &A);
    if(A < 0) {
        printf("Invalid number");
    }
    else
    {
        for (i = 1; i <= A; i++) {
            sum = sum + i;
        }
        printf("sum = %ld\n",sum);
    }

    return 0;
}
```

Παράδειγμα

Να διαβάσετε 5 ακεραίους αριθμούς από το πληκτρολόγιο. Να εκτυπώσετε το άθροισμα των αρτίων αριθμών.

Αρχή Αλγόριθμου

sum = 0

Για i = 0 ΜΕΧΡΙ 4 Με Βήμα 1

Διάβασε A

Αν $(A \% 2) = 0$ Τότε

sum = sum + A

Τέλος Αν

Τέλος Επανάληψης

Εκτύπωσε sum

Τέλος Αλγόριθμου

```
#include <stdio.h>
#include <stdlib.h>

int main(int argc, char *argv[])
{
    int i, A;
    long int sum = 0;

    for (i = 1; i <= 5; i++) {
        printf("Type an integer number : ");
        scanf("%d",&A);
        if((A % 2) == 0)
        {
            sum = sum + A;
        }
    }
    printf("sum = %d\n",sum);
    return 0;
}
```

```
Type an integer number : 7
Type an integer number : 22
Type an integer number : 3
Type an integer number : 9
Type an integer number : 10
sum = 32
```

Άσκηση

Τι θα εκτυπώσει το παρακάτω πρόγραμμα;

```
int main(int argc, char *argv[])
{
    int x = 0, sum = 0;

    for(x = 40; x > 35; x++)
    {
        sum = sum + 10;
    }

    printf("sum = %d", sum);

    return 0;
}
```


Η συνθήκη είναι συνέχεια αληθής, οπότε η εκτέλεση του προγράμματος θα συνεχίζεται μέχρι να υπερχειλίσει η μεταβλητή x και να γίνει μικρότερη του 35.

Τι θα εκτυπώσει το παρακάτω πρόγραμμα;

```
int main(int argc, char *argv[])
{
    int x = 0, y = 10, sum = 0;

    for(x = 25; (x <=30) && (y < 20); x++)
    {
        sum = sum + 10;
        y = y + 5;
    }

    printf("sum = %d", sum);

    return 0;
}
```

Το πρόγραμμα τυπώνει "sum = 20" επειδή θα γίνουν δύο επαναλήψεις.

$y = 10$

`for(x = 25; (x <=30) && (y < 20); x++)`

$x = 25$ μικρότερο του 30

$y = 10$ μικρότερο του 20

1. Αληθής $sum = sum + 10$ οπότε $sum = 10$

$x = 26$ μικρότερο του 30

$y = 15$ μικρότερο του 20

2. Αληθής $sum = sum + 10$ οπότε $sum = 20$

$x = 27$ μικρότερο του 30

$y = 20$ μικρότερο του 20

ΨΕΥΔΗΣ

Παράδειγμα

Να διαβάσετε από το πληκτρολόγιο την τιμή και την ποσότητα 3 προϊόντων. Σε κάθε επανάληψη να εκτυπώνετε το σύνολο χρημάτων που αντιστοιχεί στο κάθε προϊόν. Τέλος να εκτυπώνετε το σύνολο των χρημάτων που πρέπει να πληρώσει ο πελάτης.

```
Type quantity : 2
Type value : 3
Product = 6.000000
=====
Type quantity : 5
Type value : 3
Product = 15.000000
=====
Type quantity : 1
Type value : 10
Product = 10.000000
=====
Total = 31.000000
```

```
int main(int argc, char *argv[]) {
    float value, product, total = 0.0;
    int quantity, i;

    for(i = 0; i < 3; i++) {
        printf("Type quantity : ");
        scanf("%d", &quantity);
        printf("Type value : ");
        scanf("%f", &value);
        product = quantity * value;
        printf("Product = %f\n",product);
        total = total + product;
        printf("=====\n");
    }
    printf("Total = %f", total);
    return 0;
}
```

Ατέρμονος βρόχος

Σε αρκετά προγράμματα υπάρχει η ανάγκη να δημιουργήσουμε έναν ατέρμονα βρόγχο δηλαδή μία επανάληψη που δεν τελειώνει ποτέ. Ο λόγος για να το κάνουμε αυτό είναι επειδή δεν γνωρίζουμε το ακριβές πλήθος των επαναλήψεων. Για παράδειγμα θεωρίστε ένα πρόβλημα που είναι αναγκαίο να διαβάζουμε αριθμούς από το πληκτρολόγιο μέχρι να διαβάσουμε τον αριθμό 5. Σε αυτή την περίπτωση δεν γνωρίζουμε το ακριβές πλήθος των αριθμών, άρα θα χρειαστεί να χρησιμοποιήσουμε μία συνθήκη όπου θα είναι πάντα αληθής. Για παράδειγμα :

```
int i;  
for(i = 1; i < 3; i=i+0)
```

```
int i;  
for(i = 1; 2 < 3; i=i++)
```

Άλλες παραλλαγές

Εναλλακτικά μπορούμε να απαλείψουμε την αρχική τιμή, τη συνθήκη, και το βήμα ώστε ο ατέρμονος βρόγχος να πάρει τη μορφή :

```
for(;;) {  
    // Εντολές επανάληψης  
}
```

Εάν σε περίπτωση θέλουμε να σταματήσουμε την επανάληψη και να συνεχίσουμε εκτός του ατέρμονος βρόγχου θα πρέπει ανάμεσα στις εντολές της επανάληψης να υπάρχει το `break` το οποίο θα μεταφέρει τη ροή του προγράμματος έξω από το `for`.

Παράδειγμα ατέρμονος βρόγχου

Έστω ότι θέλουμε να εκτυπώνουμε συνέχεια δύο προτάσεις, τη "first line" και τη "second line". Εάν δεν υπήρχε μέσα στην επανάληψη η δομή if τότε το πρόγραμμα θα εκτυπώνει συνέχεια αυτές τις δύο γραμμές. Για λόγους επίδειξης του break προσθέσαμε έναν μετρητή, ο οποίος μόλις γίνει η τιμή του ίση με 3, τότε διακόπτει την επανάληψη.

```
int main(int argc, char *argv[]) {
    int i = 1;
    for(;;) {
        printf("%d) first line\n", i );
        i = i + 1;
        if(i > 3){
            break;
        }
        printf("\tsecond line\n");
    }
}
```

Όπως βλέπουμε και από το αποτέλεσμα, την τελευταία φορά (3) δεν εκτυπώνεται η πρόταση "second line" γιατί είναι μετά το break, το οποίο αγνοεί όλες επόμενες εντολές, και μεταφέρει τη ροή του προγράμματος έξω από το for.

```
1) first line
   second line
2) first line
   second line
3) first line
```