



ΠΑΝΕΠΙΣΤΗΜΙΟ
ΠΑΤΡΩΝ
UNIVERSITY OF PATRAS

ΑΝΟΙΚΤΑ ακαδημαϊκά
μαθήματα ΠΠ

Προγραμματισμός Η/Υ

Ενότητα 5: Εντολές Επανάληψης

Νίκος Καρακαπιλίδης, Καθηγητής

Δημήτρης Σαραβάνος, Καθηγητής

Πολυτεχνική Σχολή

Τμήμα Μηχανολόγων & Αεροναυπηγών Μηχανικών

Σκοποί ενότητας

- Έλεγχος της ροής του προγράμματος
- Κατανόηση της χρήσης των εντολών **while**, **do ... while** και **for**
- Κατανόηση της λειτουργίας των βρόχων μέσα στο πρόγραμμα
- Κατανόηση της χρήσης των εντολών **break**, **continue** και **goto**



Περιεχόμενα ενότητας

- Έλεγχος ροής προγράμματος (βλ. ενότητα #4)
- Προτάσεις επανάληψης
- Περί βρόχων (loops)
- Προτάσεις **break**, **continue** και **goto**



Μέρος 1^ο

Έλεγχος ροής προγράμματος

Έλεγχος ροής προγράμματος

Δομή επανάληψης

Έλεγχος ροής προγράμματος

Βασικές δομές (**ανεξάρτητες της ΓΠ** - χρησιμοποιούνται για την ανάπτυξη αλγορίθμων επίλυσης προβλημάτων)

Δομή Ακολουθίας (sequence)

- Ένα σύνολο προτάσεων (εντολών) εκτελείται ακολουθιακά (η μία μετά την άλλη)

Δομή Επιλογής (selection)

- Ανάλογα με το αν μια συνθήκη είναι αληθής ή ψευδής, εκτελούνται διαφορετικές προτάσεις (ή ομάδες προτάσεων)

Δομή Επανάληψης (loop, repetition)

- Μια πρόταση (η μια ομάδα προτάσεων) εκτελείται πολλές φορές, είτε όσο μια συνθήκη είναι αληθής είτε για προκαθορισμένο αριθμό επαναλήψεων

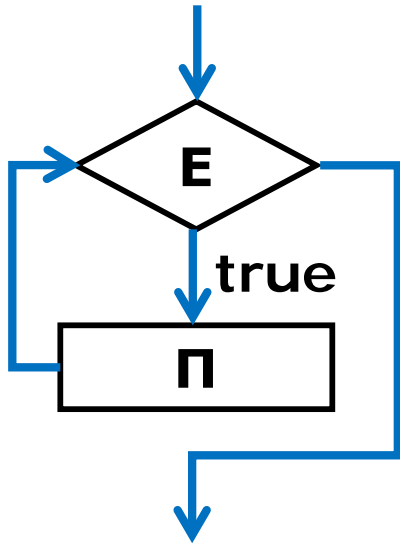
Μέρος 2^ο

Προτάσεις επανάληψης

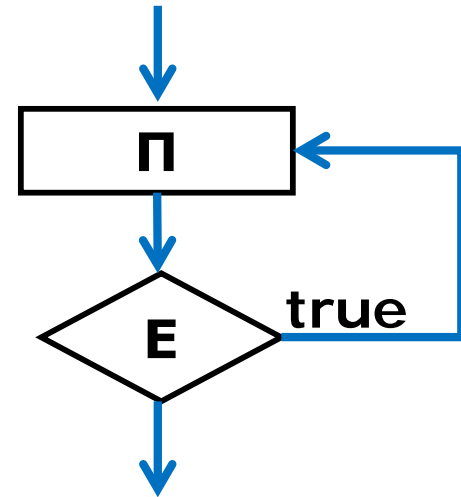
Προτάσεις επανάληψης

Εντολές **while**, **do ... while** και **for**

Προτάσεις επανάληψης



Με συνθήκη εισόδου
while E do Π



Με συνθήκη εξόδου
do Π while E
repeat Π until E



Προτάσεις επανάληψης στη C

Πρόταση **while**

Εκτελεί μια πρόταση ή μια ομάδα προτάσεων όσο μια έκφραση (συνθήκη) είναι αληθής

**while (έκφραση)
πρόταση;**

```
int x=0;  
...  
while(x<5){  
    printf("%d\n",x);  
    ++x;  
}
```

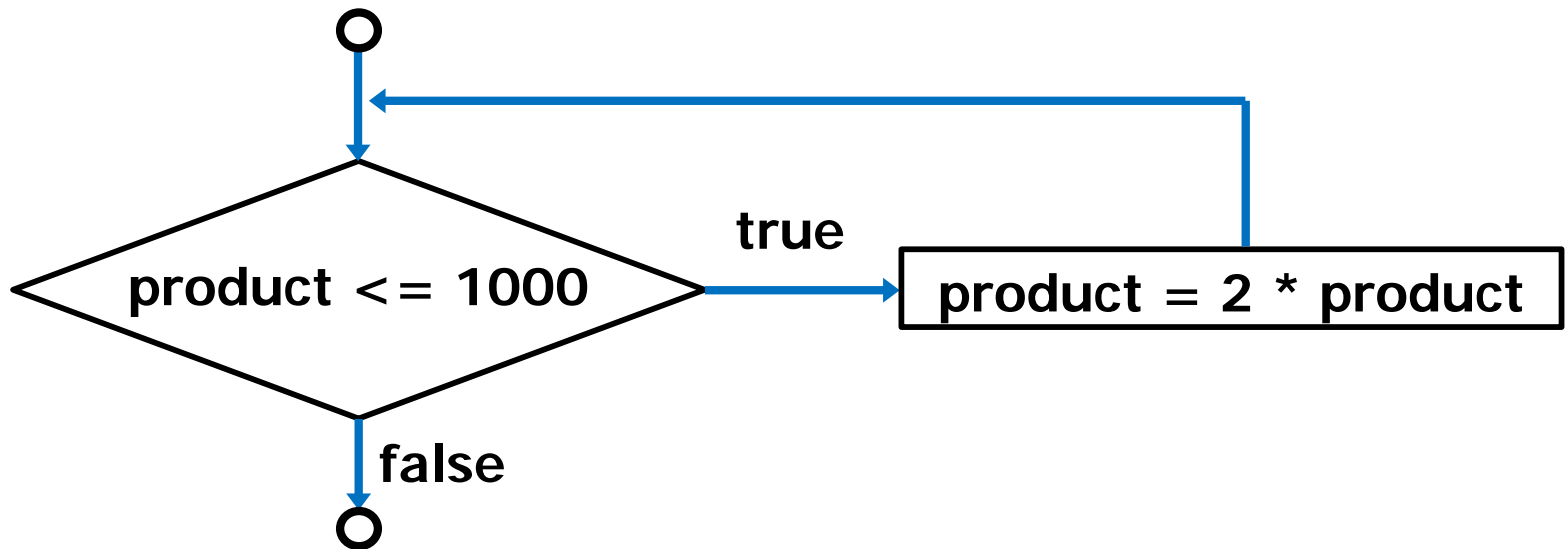
**while (έκφραση) {
 πρόταση_1;
 πρόταση_2;
 ...
 πρόταση_n;
}**

block του while



Προτάσεις επανάληψης στη C

```
...  
int product = 2;  
...  
while ( product <= 1000 )  
    product = 2 * product;  
...
```



Προτάσεις επανάληψης στη C

Πρόταση **do ... while**

Εκτελεί μια πρόταση (ή μια ομάδα προτάσεων) και στη συνέχεια, για όσο μια έκφραση είναι αληθής, επαναλαμβάνει την εκτέλεση της πρότασης (ή της ομάδας προτάσεων)

```
do
    πρόταση
while (έκφραση) ;
```

```
int count=0, limit=5;
...
do {
    count++;
    printf("count is %d\n",count);
}
while (count<limit);
```

```
do {
    πρόταση_1;
    πρόταση_2
    ...
    πρόταση_n
}
while (έκφραση);
```

Προτάσεις επανάληψης στη C

```
1 /* Using the do/while repetition structure */
2 #include <stdio.h>
3 main()
4 {
5     int counter = 1;
6     do {
7         printf( "%d", counter );
8     } while ( ++counter <= 10 );
9 }
```



Προτάσεις `while` και `do ... while`

```
int number;  
printf("Enter a positive value: ");  
scanf("%d",&number);  
while(number <= 0){  
    printf("Enter a positive value: ");  
    scanf("%d",&number);  
}
```

```
int number;  
do{  
    printf("Enter a positive value: ");  
    scanf("%d",&number);  
}  
while(number <= 0);
```

Προσοχή: Το σώμα εντολών της `while` μπορεί να μην εκτελεστεί ποτέ, ενώ το σώμα εντολών της `do-while` θα εκτελεστεί τουλάχιστον μία φορά



Προτάσεις επανάληψης στη C

Πρόταση **for**

Ειδική περίπτωση επανάληψης εκτέλεσης μιας πρότασης ή μιας ομάδας προτάσεων

```
for (init;test;update){  
    πρόταση;  
}
```

init → αρχικοποίηση
test → έλεγχος συνθήκης
update → ενημέρωση

```
int x;  
for(x=0;x<5;++x){  
    printf("%d\n",x);  
}
```

```
for (init;test;update) {  
    πρόταση_1;  
    πρόταση_2;  
    ...  
    πρόταση_n;  
}
```

block του for



Προτάσεις επανάληψης στη C

```
int x;  
for(x=0;x<5;++x){  
    printf("%d\n",x);  
}
```

| x | x<5 | output |
|---|-----|--------|
| 0 | 1 | 0 |
| 1 | 1 | 1 |
| 2 | 1 | 2 |
| 3 | 1 | 3 |
| 4 | 1 | 4 |
| 5 | 0 | |



Προτάσεις επανάληψης στη C

```
1 /* Summation with for */
2 #include <stdio.h>
3
4 main()
5 {
6     int sum = 0, number;
7
8     for ( number = 2; number <= 100; number += 2 ) {
9         sum += number;
10        }
11        printf( "Sum is %d\n", sum );
12 }
```

Sum is 2550



output προγράμματος



Προτάσεις while και for

```
int x;  
x=0;  
while(x<5){  
    printf("%d\n",x);  
    ++x;  
}
```

```
int x;  
for(x=0;x<5;++x){  
    printf("%d\n",x);  
}
```

Προσοχή:

Οποιαδήποτε δομή **while** μπορεί να εκφραστεί (γραφεί) ως δομή **for** και το αντίθετο



Μέρος 3^ο

Περί βρόχων (loops)

Περί βρόχων (loops)

Τελεστές, Μετρητές, Σημαίες

Περί βρόχων

Τελεστές (βλ. ενότητα #2)

Μοναδιαίοι αύξησης και μείωσης ++, --

- Προπορευόμενοι (prefix), π.χ. ++**x**

→ η αύξηση ή η ελάττωση γίνεται πριν να χρησιμοποιηθεί η τιμή της μεταβλητής στον υπολογισμό της τιμής της έκφρασης στην οποία συμμετέχει



Περί βρόχων

- Παρελκόμενοι (postfix), π.χ. **x++**
- η αύξηση ή η ελάττωση γίνεται μετά τον υπολογισμό της τιμής της έκφρασης στην οποία συμμετέχει

```
i = 5;  
x = ++i;  
y = i++;
```

τελικά
→

| x | y | i |
|----------|----------|----------|
| 6 | 6 | 7 |



Περί βρόχων

```
int x;  
x = 0;  
while(x < 5){  
    printf("%d\n", x);  
    ++x;  
}
```

Δήλωση μετρητή (μεταβλητής που θα χρησιμοποιηθεί στη συνέχεια για τον έλεγχο των επαναλήψεων)

Αρχικοποίηση μετρητή

Συνθήκη ελέγχου επανάληψης

Αύξηση «βήματος» μετρητή



Περί βρόχων

Χρήση «σημαίας» (σε **while**)

Παράδειγμα: άθροισμα σειράς αριθμών απροσδιόριστου μεγέθους (τερματίζεται όταν ο χρήστης εισάγει το 0)

```
int number, sum=0;
scanf("%d",&number);      /* διάβασε τον πρώτο αριθμό */
while(number != 0){       /* συνθήκη ελέγχου επανάληψης */
    sum += number;        /* sum=sum+number */
    scanf("%d",&number); /* διάβασε τον επόμενο αριθμό */
}
printf("The sum is %d\n",sum);
```




Περί βρόχων

Φωλιασμένοι βρόχοι

```
int i=0,j;  
while(i<5){  
    j=0;  
    while(j<5){  
        printf("%d-%d",i,j);  
        ++j;  
    }  
    printf("\n");  
    ++i;  
}
```

output
προγράμματος



```
0-0 0-1 0-2 0-3 0-4  
1-0 1-1 1-2 1-3 1-4  
2-0 2-1 2-2 2-3 2-4  
3-0 3-1 3-2 3-3 3-4  
4-0 4-1 4-2 4-3 4-4
```

Περί βρόχων

```
int i=0,j;  
while(i<5){  
    j=0;  
    while(j<=i){  
        printf("%d-%d",i,j);  
        ++j;  
    }  
    printf("\n");  
    ++i;  
}
```

output
προγράμματος

0-0
1-0 1-1
2-0 2-1 2-2
3-0 3-1 3-2 3-3
4-0 4-1 4-2 4-3 4-4

Μέρος 4^ο

Προτάσεις **break**, **continue** και **goto**

Προτάσεις **break**, **continue** και **goto**

Σύνταξη των προτάσεων στο πρόγραμμα

Διαχείριση ειδικών περιπτώσεων

Η πρόταση **break**

Σε πρόταση επανάληψης, μεταφέρει τον έλεγχο έξω από τον βρόχο στον οποίο εμπεριέχεται (στην πρόταση που ακολουθεί τον βρόχο)

Χρήση της στην πρόταση **switch** (βλ. ενότητα #4)

- Αναντικατάστατη

Σε άλλες περιπτώσεις, υπάρχει τρόπος να γραφεί κώδικας χωρίς τη χρήση της

- Πολλές **break** καταστρέφουν τη δόμηση του κώδικα



Διαχείριση ειδικών περιπτώσεων

```
while (έκφραση) {  
    if (ειδική περίπτωση) {  
        προτάσεις επεξεργασίας ειδικής περίπτωσης  
        break;  
    }  
    προτάσεις επεξεργασίας κανονικών περιπτώσεων  
}
```



Διαχείριση ειδικών περιπτώσεων

Η πρόταση **continue**

Μεταφέρει τον έλεγχο στην αρχή του βρόχου, παραλείποντας την εκτέλεση του υπολοίπου τμήματος του σώματος του βρόχου



Διαχείριση ειδικών περιπτώσεων

```
while (έκφραση) {  
    if (κανονική περίπτωση) {  
        προτάσεις επεξεργασίας κανονικής περίπτωσης  
        continue;  
    }  
    προτάσεις επεξεργασίας ειδικών περιπτώσεων  
}
```

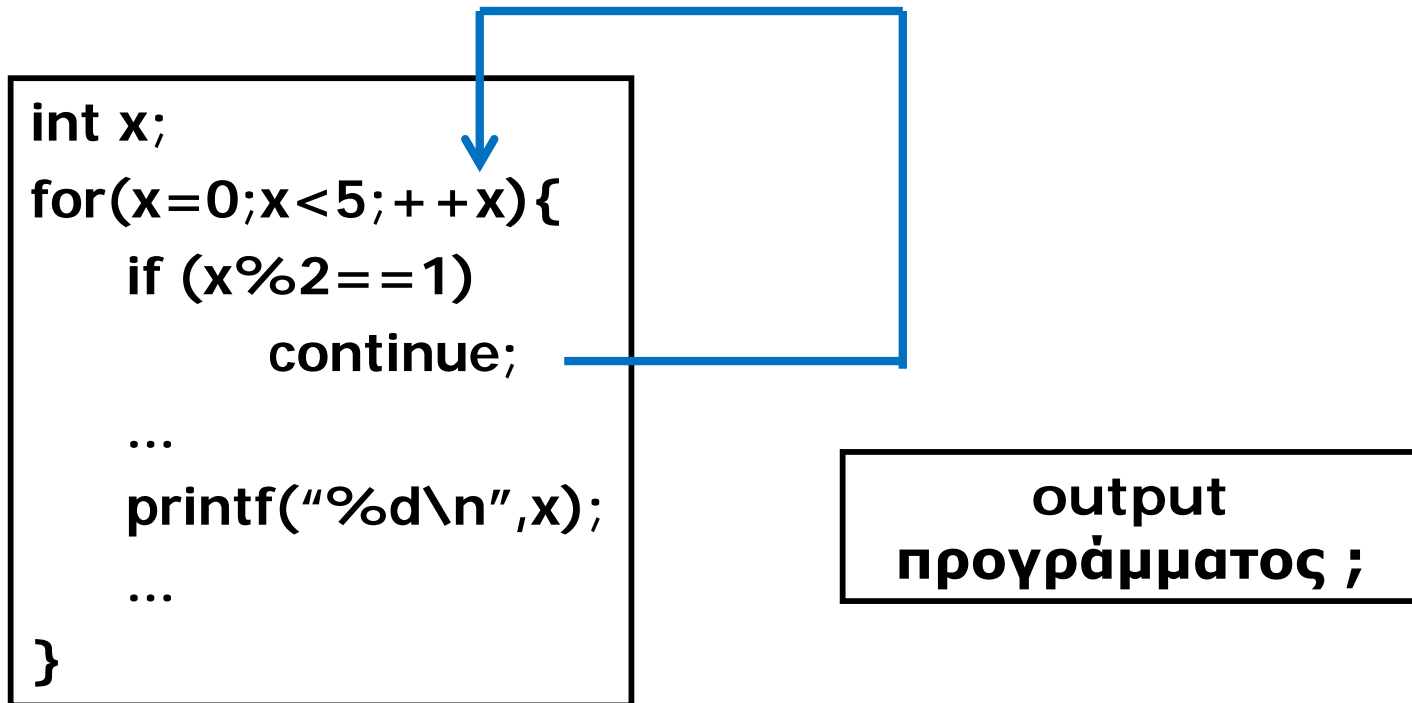
**Αποφυγή
continue**

```
while (έκφραση) {  
    if (κανονική περίπτωση)  
        προτάσεις επεξεργασίας κανονικής περίπτωσης  
    else  
        προτάσεις επεξεργασίας ειδικών περιπτώσεων  
}
```



Διαχείριση ειδικών περιπτώσεων

Παράδειγμα



Διαχείριση ειδικών περιπτώσεων

Η πρόταση **goto**

Σύνταξη: **goto <ετικέτα>;**

- Μεταφέρει τον έλεγχο στην πρόταση που σημειώνεται με την ετικέτα ως **<ετικέτα>: πρόταση**

Προσοχή: να την αποφεύγετε

- Καταστρέφει τη δόμηση του κώδικα
- Δεν είναι απαραίτητη → μπορείτε να γράψετε οποιοδήποτε πρόγραμμα χωρίς τη χρήση της



Λάθη που γίνονται συχνά

Χρήση **if** αντί για **while** και αντίστροφα

Παράλειψη παρένθεσης στη συνθήκη (ισχύει για τις προτάσεις **while** και **do...while**)

- Π.χ. **while x > 4** αντί του ορθού **while (x > 4)**

Παράλειψη ; στην πρόταση **for**

Παράλειψη αγκυλών στο σώμα του **while**

Χρήση τελεστών **++**, **--** καθώς και σύνθετων τελεστών ανάθεσης (**+=** κλπ.), ιδιαίτερα σε πολύπλοκες εκφράσεις



Χρήση προτάσεων ροής ελέγχου

Γενικοί κανόνες

Προσοχή στην αναγνωσιμότητα του κώδικα

- Χρήση tabs
- Χρήση αγκυλών

Αποφύγετε το **goto**

- Πολλές φορές δείχνει αδυναμία του προγραμματιστή να αναπτύξει δομημένο κώδικα



Χρήση προτάσεων ροής ελέγχου

Προτιμήστε προτάσεις **while** (βρόχος επανάληψης με συνθήκη εισόδου) από το **do...while** (βρόχος επανάληψης με συνθήκη εξόδου)

- Ο κώδικας είναι πιο ευανάγνωστος

Χρησιμοποιήστε την εντολή **break** σε προτάσεις **switch**

- Αποφύγετε όμως τη χρήση των **break** και **continue** σε βρόχους επανάληψης

Βεβαιωθείτε ότι ο βρόχος επανάληψης που αναπτύξατε θα τερματιστεί μετά από πεπερασμένο αριθμό επαναλήψεων

- Ελέγξτε διεξοδικά όλες τις περιπτώσεις



Τέλος Ενότητας

Χρηματοδότηση

- Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στο πλαίσιο του εκπαιδευτικού έργου των διδασκόντων.
- Το έργο «**Ανοικτά Ακαδημαϊκά Μαθήματα στο Πανεπιστήμιο Πατρών**» έχει χρηματοδοτήσει μόνο την αναδιαμόρφωση του εκπαιδευτικού υλικού.
- Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους.



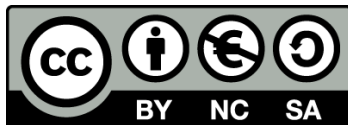
Σημείωμα Αναφοράς

Copyright Πανεπιστήμιο Πατρών, Πολυτεχνική Σχολή, Τμήμα Μηχανολόγων & Αεροναυπηγών Μηχανικών, Νίκος Καρακαπιλίδης, Δημήτρης Σαραβάνος. Νίκος Καρακαπιλίδης, Δημήτρης Σαραβάνος. «Προγραμματισμός Η/Υ. Εντολές Επανάληψης». Έκδοση: 1.0. Πάτρα 2014. Διαθέσιμο από τη δικτυακή διεύθυνση: <https://eclass.upatras.gr/courses/MECH1207/>



Σημείωμα Αδειοδότησης

Το παρόν υλικό διατίθεται με τους όρους της άδειας χρήσης Creative Commons Αναφορά, Μη Εμπορική Χρήση Παρόμοια Διανομή 4.0 [1] ή μεταγενέστερη, Διεθνής Έκδοση. Εξαιρούνται τα αυτοτελή έργα τρίτων π.χ. φωτογραφίες, διαγράμματα κ.λ.π., τα οποία εμπεριέχονται σε αυτό και τα οποία αναφέρονται μαζί με τους όρους χρήσης τους στο «Σημείωμα Χρήσης Έργων Τρίτων».



[1] <http://creativecommons.org/licenses/by-nc-sa/4.0/>

Ως **Μη Εμπορική** ορίζεται η χρήση:

- που δεν περιλαμβάνει άμεσο ή έμμεσο οικονομικό όφελος από την χρήση του έργου, για το διανομέα του έργου και αδειοδόχο
- που δεν περιλαμβάνει οικονομική συναλλαγή ως προϋπόθεση για τη χρήση ή πρόσβαση στο έργο
- που δεν προσπορίζει στο διανομέα του έργου και αδειοδόχο έμμεσο οικονομικό όφελος (π.χ. διαφημίσεις) από την προβολή του έργου σε διαδικτυακό τόπο

Ο δικαιούχος μπορεί να παρέχει στον αδειοδόχο ξεχωριστή άδεια να χρησιμοποιεί το έργο για εμπορική χρήση, εφόσον αυτό του ζητηθεί.



Διατήρηση Σημειωμάτων

Οποιαδήποτε αναπαραγωγή ή διασκευή του υλικού θα πρέπει να συμπεριλαμβάνει:

- το Σημείωμα Αναφοράς
- το Σημείωμα Αδειοδότησης
- τη δήλωση Διατήρησης Σημειωμάτων
- το Σημείωμα Χρήσης Έργων Τρίτων (εφόσον υπάρχει)

μαζί με τους συνοδευόμενους υπερσυνδέσμους.



Σημείωμα Χρήσης Έργων Τρίτων

Οποιοδήποτε έργο στην παρούσα ενότητα, έχει δημιουργηθεί από τους διδάσκοντες του μαθήματος ή/και την Τμηματική Ομάδα Εργασίας και παρέχεται με την ίδια άδεια CC BY-NC-SA 4.0

