



**Τμήμα Μηχανικών Η/Υ & Πληροφορικής
Πανεπιστήμιο Πατρών**

Εισαγωγή στον Προγραμματισμό

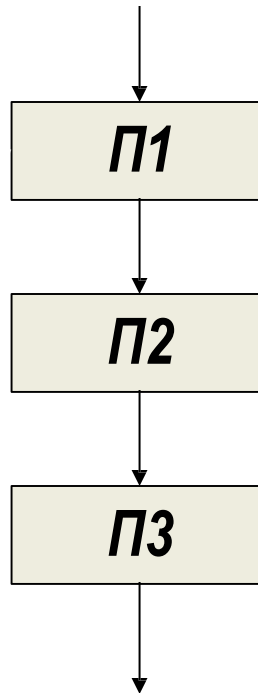
Η γλώσσα προγραμματισμού C

Δομή διάλεξης

- ▣ Έλεγχος ροής
- ▣ Πρόταση επιλογής (if, switch)
- ▣ Προτάσεις επανάληψης (while, do-while, for)
- ▣ Διακλάδωση χωρίς συνθήκη (break, continue, goto)

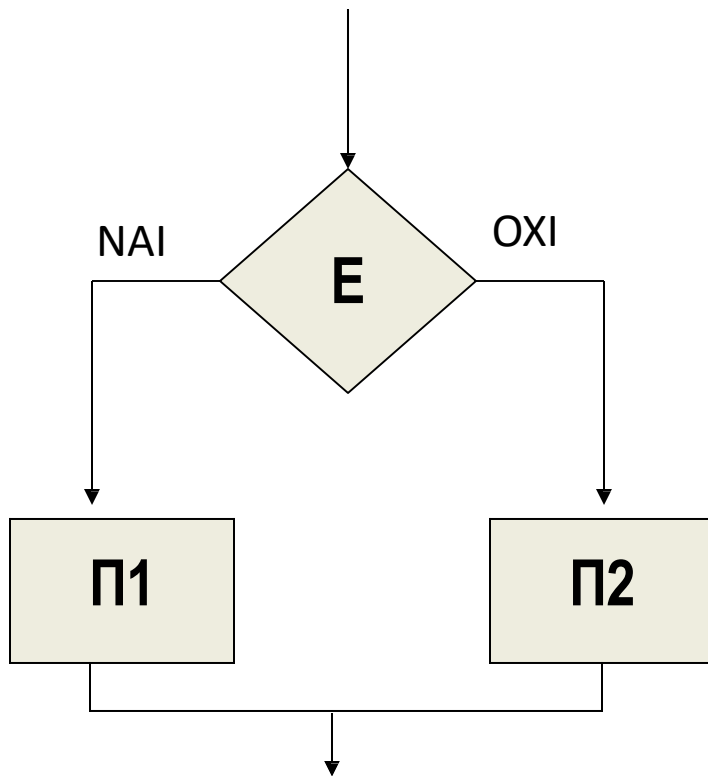
Έλεγχος Ροής

- Κανονικά, οι «προτάσεις» (εντολές) εκτελούνται σειριακά, κατ' ακολουθία



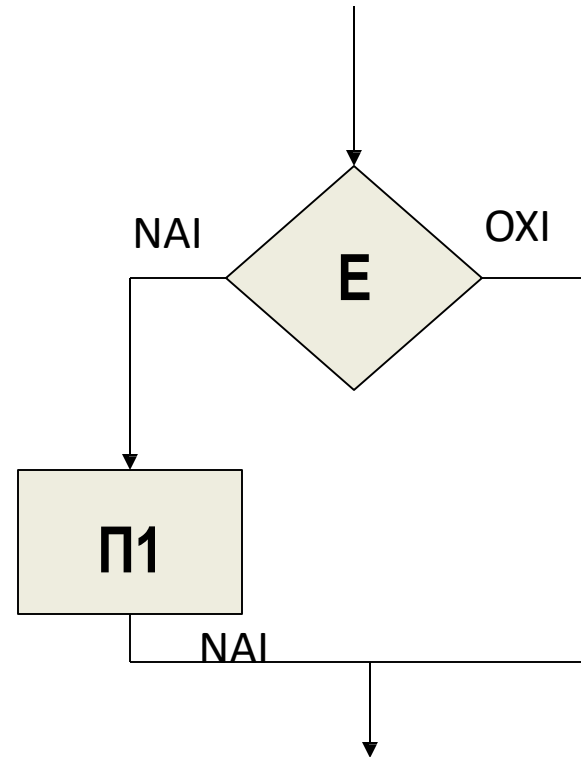
if: Πρόταση επιλογής

Απλή



```
if (E)
  Π1
else
  Π2
```

Περιορισμένη

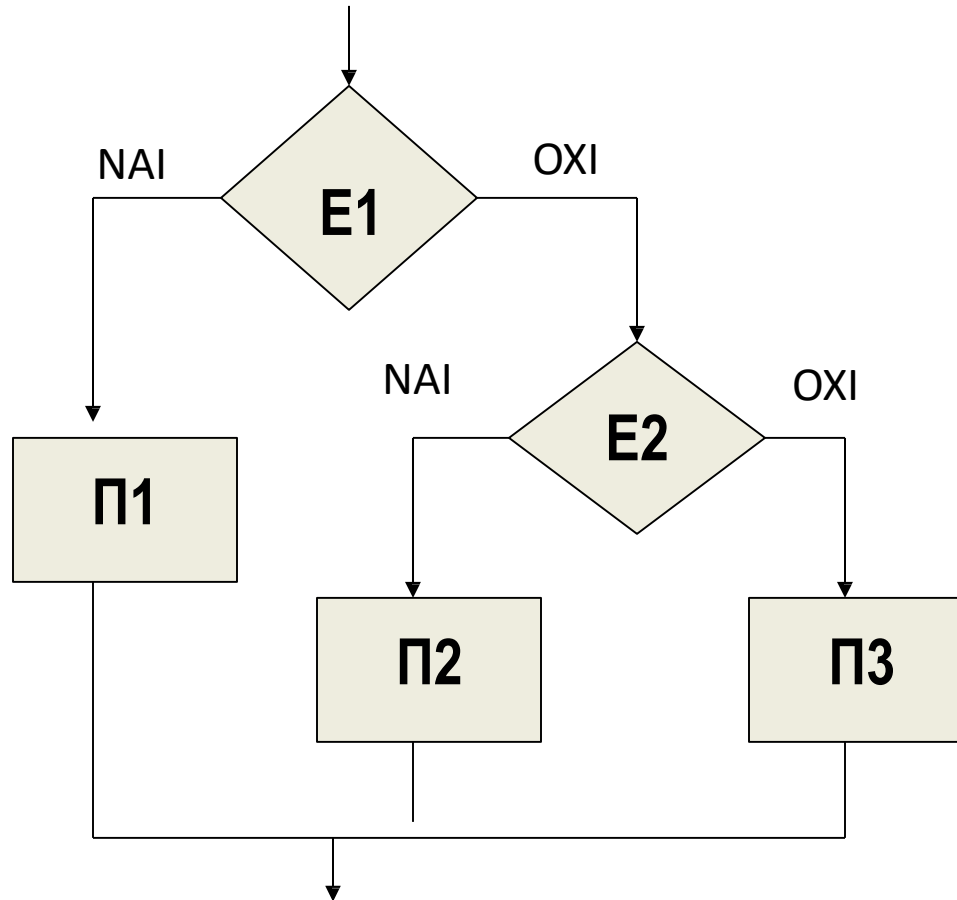


```
if
  (E)
  Π1
```

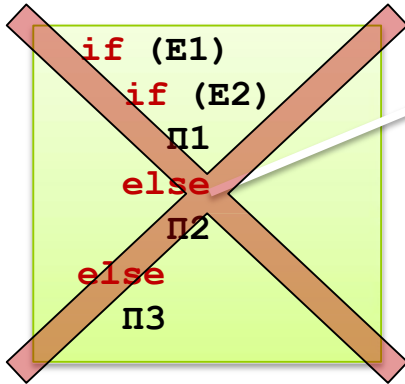
if: Δομή επιλογής

Εμφωλευμένη

```
if (E1)  
  Π1  
else if (E2)  
  Π2  
else  
  Π3
```

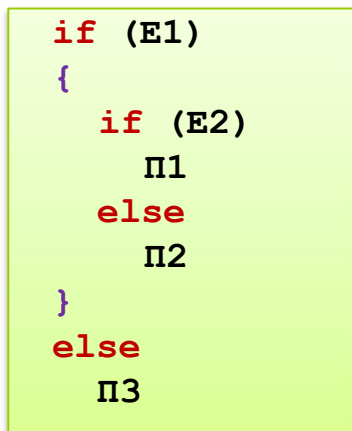


if: Δομή επιλογής

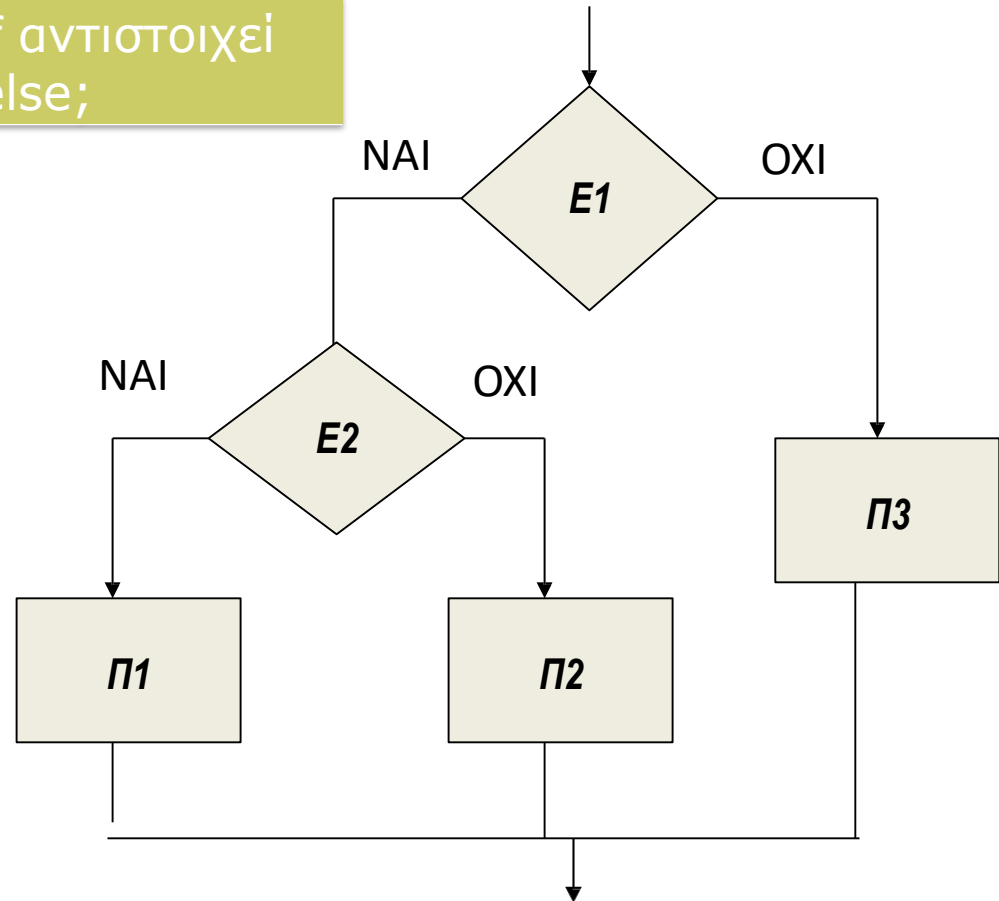


Σε ποιο if αντιστοιχεί
αυτό το else;

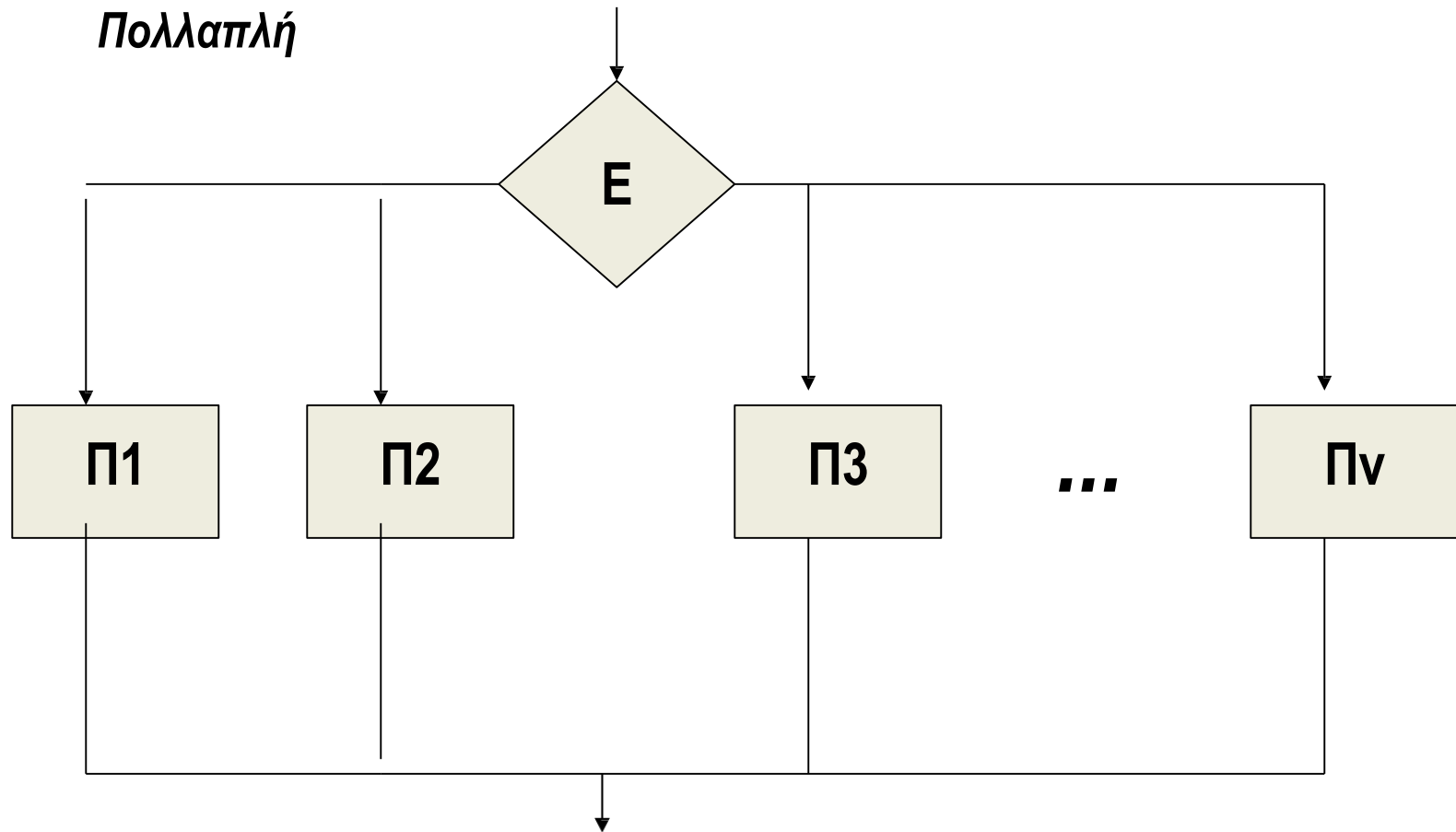
ΠΡΟΣΟΧΗ!!!
Κάποιο πρόβλημα!!!



Εμφωλευμένο



if: Δομή επιλογής



if: Δομή επιλογής

if (<έκφραση>
 <πρόταση1>
[else <πρόταση2>]

απλή ή σύνθετη

If (<έκφραση>
 <πρόταση1>
 else if (<έκφραση2>
 <πρόταση2>
 else if (<έκφραση3>
 <πρόταση3>
 else <πρόταση4>

Εμφωλευμένο if

Παράδειγμα

```
#include <stdio.h>

main()
{
    float num;
    printf("Δώσε αριθμό: ");
    scanf("%f", &num);

    if (num < 0)
        printf("Η απόλυτη τιμή του %f είναι: %f\n", num, -num);
    else
        printf("Η απόλυτη τιμή του %f είναι: %f\n", num, num);

    printf("Η απόλυτη τιμή: %f\n", num < 0 ? -num : num);
}
```

switch: Πολλαπλή επιλογή

- Όταν βάσει μιας έκφρασης θέλουμε να επιλέξουμε ανάμεσα σε πολλές επιλογές, είναι πιο «βολική» η χρήση της **switch**.

```
switch (<έκφραση>
{
    case <σταθ-εκφρ-1>: <προτ-1>; break;
    case <σταθ-εκφρ-2>: <προτ-2>; break;
    ...
    case <σταθ-εκφρ-N>: <προτ-N>; break;
    default: <πρόταση>; break;
}
```

switch: Πολλαπλή επιλογή

- Κάθε <σταθ-έκφρ-ι> πρέπει να είναι **μία τιμή int ή char** ή μία έκφραση μόνο με τέτοιες τιμές
- Δύο <σταθ-εκφρ-ι> δεν μπορούν να έχουν την ίδια τιμή
- Αν <έκφραση>=<σταθ-εκφρ-χ> τότε εκτελούνται όλες οι παρακάτω της x προτάσεις
 - ✓ Για να το αποτρέψουμε αυτό, χρειάζεται η **break**!
- Η <πρόταση> εκτελείται μόνο όταν καμιά από τις <προτ-ι> δεν ικανοποιείται
- Δεν υποστηρίζονται περιοχές τιμών (ranges), μόνο ισότητα
- Η default δεν είναι απαραίτητο να είναι στο τέλος

Παράδειγμα

```
switch (choice) {  
    case 1:  
        x=a+b;  
        break;  
    case 2:  
        x=a-b;  
        break;  
    case 3:  
        x=a*b;  
        break;  
    case 4:  
        x=a/b;  
        break;  
    default:  
        printf("Ανύπαρκτη επιλογή");  
        break;  
}
```

Παράδειγμα

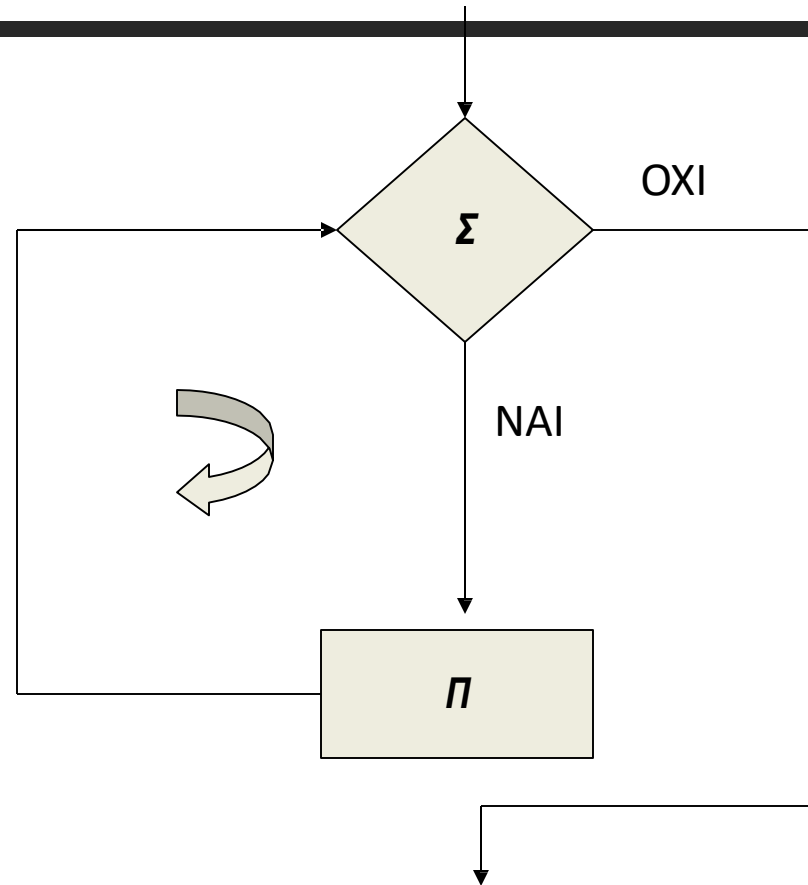
```
switch (choice) {  
    case 1: x=a+b; break;  
    case 2: x=a-b; break;  
    case 3: x=a*b;break;  
    case 4: x=a/b;break;  
    default:  
        printf("Ανύπαρκτη επιλογή");  
        break;  
}
```

Loops (Βρόχοι)

- Επανάληψη τμήματος κώδικα
 - ✓ για έναν αριθμό επαναλήψεων
 - ✓ ή γενικά μέχρι να ισχύσει μία συνθήκη

while: Loop με Συνθήκη Εισόδου

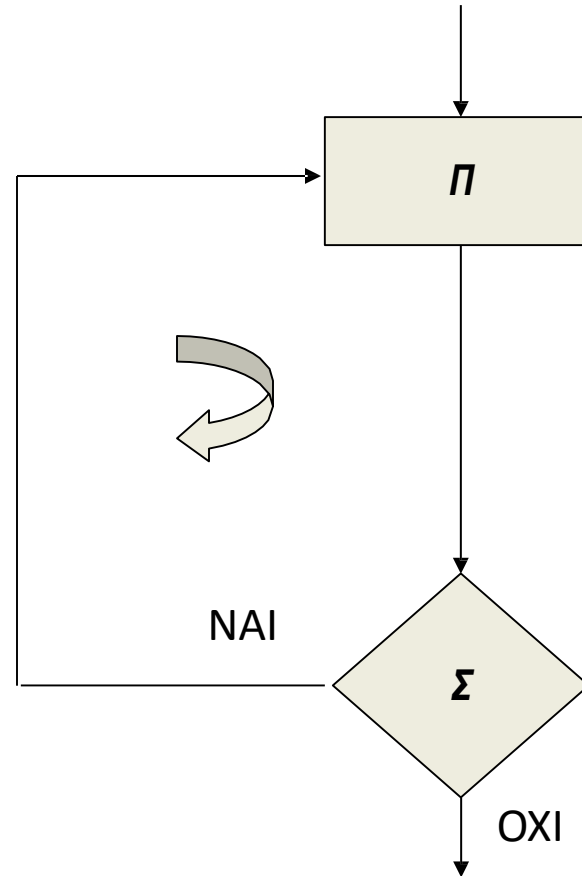
```
while (Σ)  
    Π
```



do-while: Loop με Συνθήκη Εξόδου

```
do  
  Π  
while (Σ)
```

Το Π θα εκτελεστεί
τουλάχιστον μία φορά



Loops με μετρητή

- ❑ Πολλές φορές θέλουμε ένα loop να εκτελεστεί για έναν συγκεκριμένο αριθμό επαναλήψεων
- ❑ Χρησιμοποιούμε έναν μετρητή

```
// Για να μετρήσουμε από το 0 ως το 9
```

```
int counter = 0;
```

ΑΡΧΙΚΟΠΟΙΗΣΗ

```
while (counter < 10)
```

ΣΥΝΘΗΚΗ

```
{
```

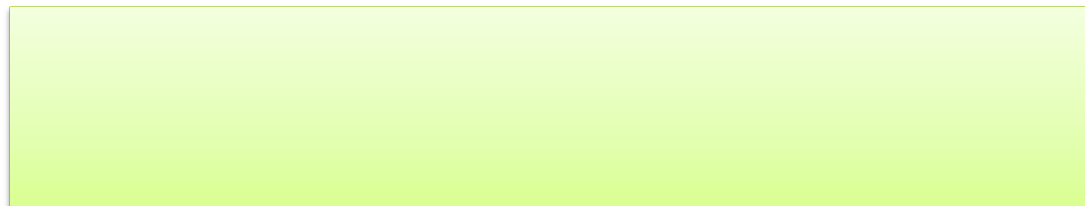
```
    printf("Ο counter είναι %d\n", counter);
```

```
    counter++;
```

ΕΝΗΜΕΡΩΣΗ

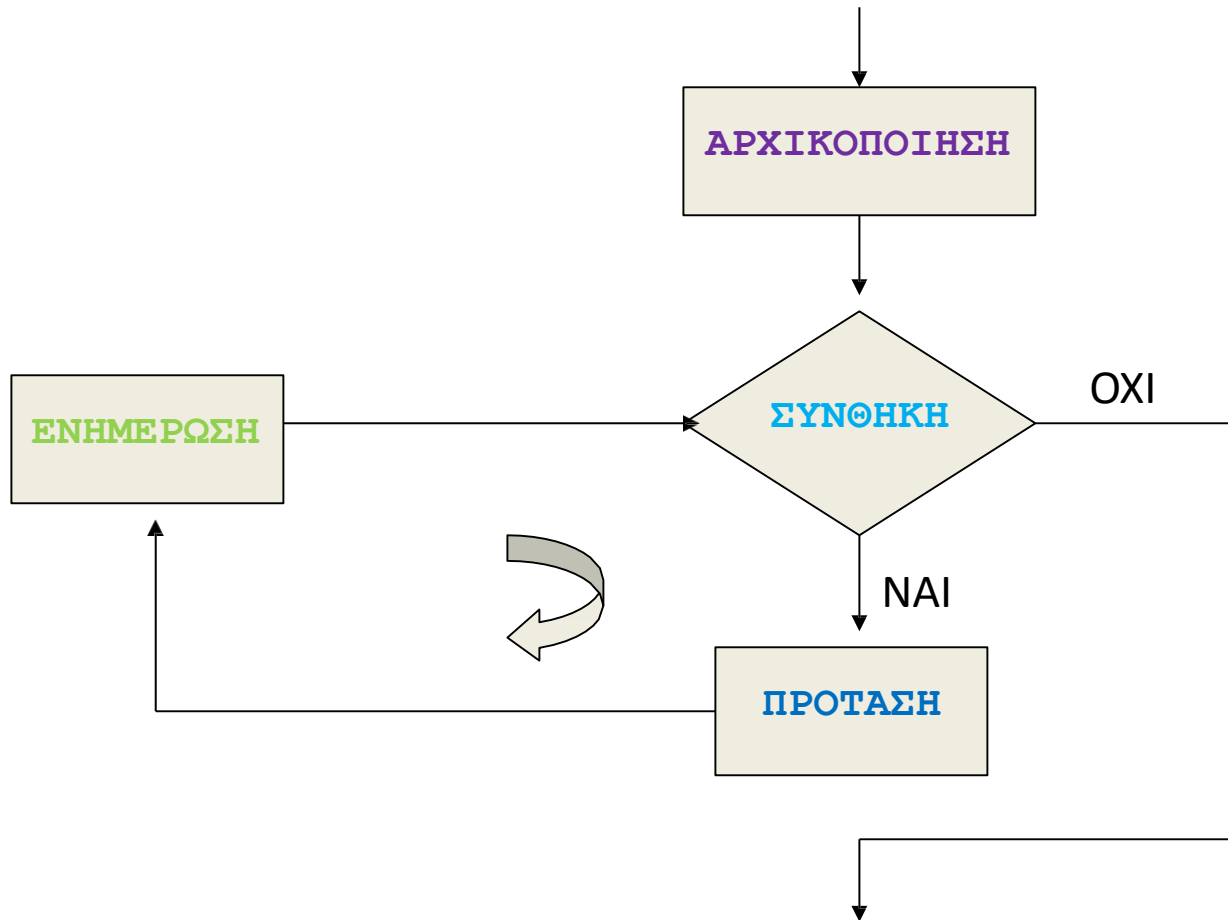
```
}
```

- ❑ Πιο εύκολα με την εντολή **for**



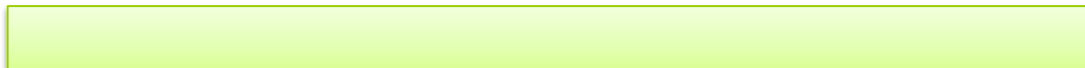
for: Loop με μετρητή

```
for (ΑΡΧΙΚΟΠΟΙΗΣΗ; ΣΥΝΘΗΚΗ; ΕΝΗΜΕΡΩΣΗ)  
    ΠΡΟΤΑΣΗ
```



for: Loop με μετρητή

- Απαρίθμηση από 0 ως $n-1$: `for (i=0; i<n; i++)`
- Απαρίθμηση από 1 ως n : `for (i=1; i<=n; i++)`
- Απαρίθμηση από $n-1$ ως 0: `for (i=n-1; i>=0; i--)`
- Απαρίθμηση από n ως 1: `for (i=n; i>=1; i--)`
- Δυνάμεις του 2 μέχρι $1e6$: `for (i=1; i<1e6; i*=2)`



Επιλογή Δομής Επανάληψης

- ❑ Προτιμούμε τη δομή με συνθήκη εισόδου (while) από αυτή με συνθήκη εξόδου (do-while)
- ❑ Προτιμούμε τη for από τη while, αν υπάρχει (ή μπορεί να οριστεί) απαριθμητής που συνοδεύεται από αρχικοποίηση και ανανέωση της τιμής του

Διακλάδωση χωρίς συνθήκη

- Διαχείριση ειδικών περιπτώσεων σε προτάσεις επανάληψης
 - ✓ break
 - ✓ continue
- Ρητή διακλάδωση
 - ✓ goto <ετικέτα>

break: Διακοπή loop

- Την είδαμε στη switch
- Γενικά, προκαλεί την έξοδο μόνο από τον πιο εσωτερικό βρόχο
- Προσοχή!
 - ✓ Καταστρέφει τη δόμηση του κώδικα
 - ✓ Υπάρχει πάντα τρόπος να γραφεί κώδικας ~~χωρίς~~ τη χρήσης

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

continue: Παράκαμψη επανάληψης

- ❑ Παρακάμπτει την τρέχουσα επανάληψη, περνώντας άμεσα στην επόμενη
- ❑ Επηρεάζει μόνο τον πιο εσωτερικό βρόχο
- ❑ Προσοχή!
 - ✓ Καταστρέφει τη δόμηση του κώδικα
 - ✓ Υπάρχει πάντα τρόπος να γραφεί κώδικας χωρίς τη χρήσης

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

goto: Αυθαίρετη διακλάδωση

- ❑ Μεταφέρει τη ροή σε αυθαίρετο σημείο του κώδικα
- ❑ Προσοχή!
 - ✓ Πολύ επικίνδυνη
 - ✓ Πολύ κακή τεχνική
 - ✓ «Απαγορεύεται» η χρήση της!! 😊

```
...  
start:  
...  
...  
...  
goto start;  
...
```