



ΠΑΝΕΠΙΣΤΗΜΙΟ
ΠΑΤΡΩΝ
UNIVERSITY OF PATRAS

ΑΝΟΙΚΤΑ ακαδημαϊκά
μαθήματα ΠΠ

Προγραμματισμός Η/Υ

Ενότητα 4: Εντολές Επιλογής

Νίκος Καρακαπιλίδης, Καθηγητής

Δημήτρης Σαραβάνος, Καθηγητής

Πολυτεχνική Σχολή

Τμήμα Μηχανολόγων & Αεροναυπηγών Μηχανικών

Σκοποί ενότητας

- Έλεγχος της ροής ενός προγράμματος
- Κατανόηση της χρήσης των προτάσεων **if** και **if ... else**
- Κατανόηση της χρήσης των προτάσεων **switch** και **break**



Περιεχόμενα ενότητας

- Βασικές Δομές Γλωσσών Προγραμματισμού (ΓΠ)
- Προτάσεις διακλάδωσης υπό συνθήκη
- Έκφραση συνθήκης
- Οι προτάσεις **switch** και **break**



Μέρος 1^ο

Βασικές Δομές Γλωσσών Προγραμματισμού (ΓΠ)

Βασικές Δομές Γλωσσών Προγραμματισμού (ΓΠ)

Δομές ελέγχου ροής προγράμματος

Βασικές Δομές ΓΠ

Αποθήκευση και ανάκληση πληροφορίας

- Σταθερές
- Μεταβλητές
- Τύποι δεδομένων

Επεξεργασία πληροφορίας

- Τελεστές



Βασικές Δομές ΓΠ

Έλεγχος ροής προγράμματος

- Ακολουθιακή εκτέλεση εντολών
- **Εντολές επιλογής**
- Εντολές επανάληψης

Αναπαράσταση διαδικασιών

- Συναρτήσεις



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)

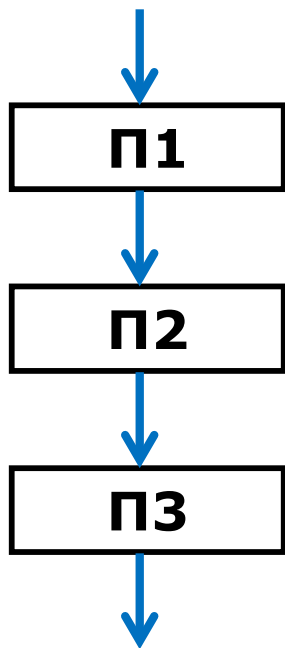
- Ανάλογα με το αν μια συνθήκη είναι αληθής ή ψευδής, εκτελούνται διαφορετικές προτάσεις (ή ομάδες προτάσεων)

Δομή Επανάληψης (loop, repetition)

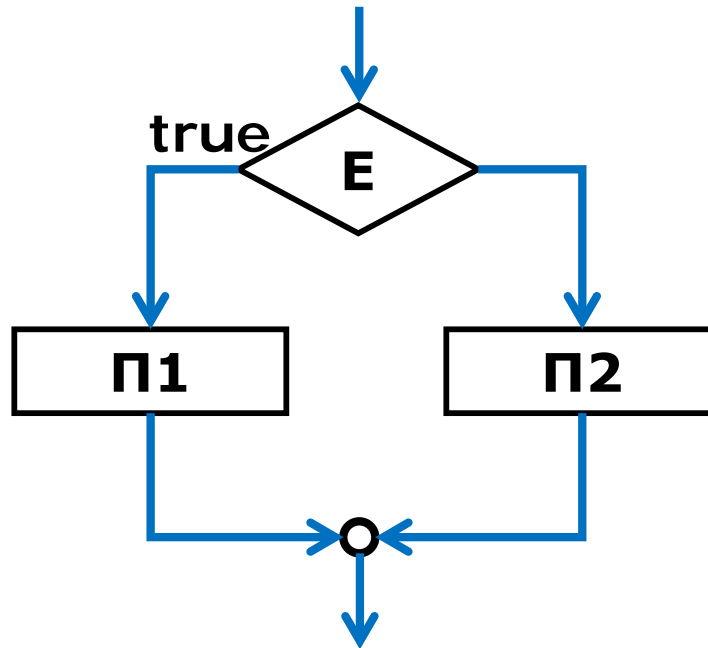
- Μια πρόταση (η μια ομάδα προτάσεων) εκτελείται πολλές φορές είτε όσο μια συνθήκη είναι αληθής είτε για προκαθορισμένο αριθμό επαναλήψεων



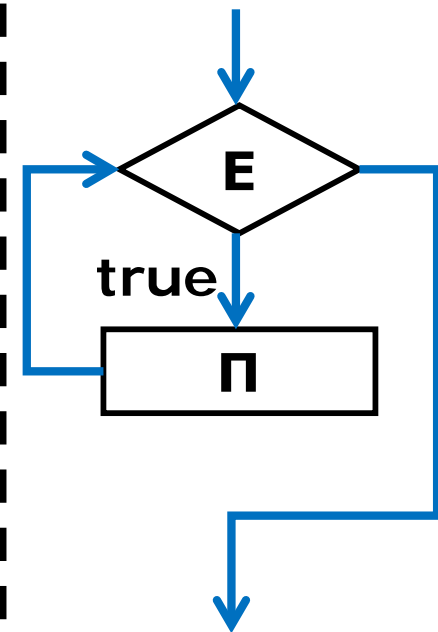
Έλεγχος ροής προγράμματος



ακολουθία



επιλογή



επανάληψη



Μέρος 2^ο

Προτάσεις διακλάδωσης υπό συνθήκη

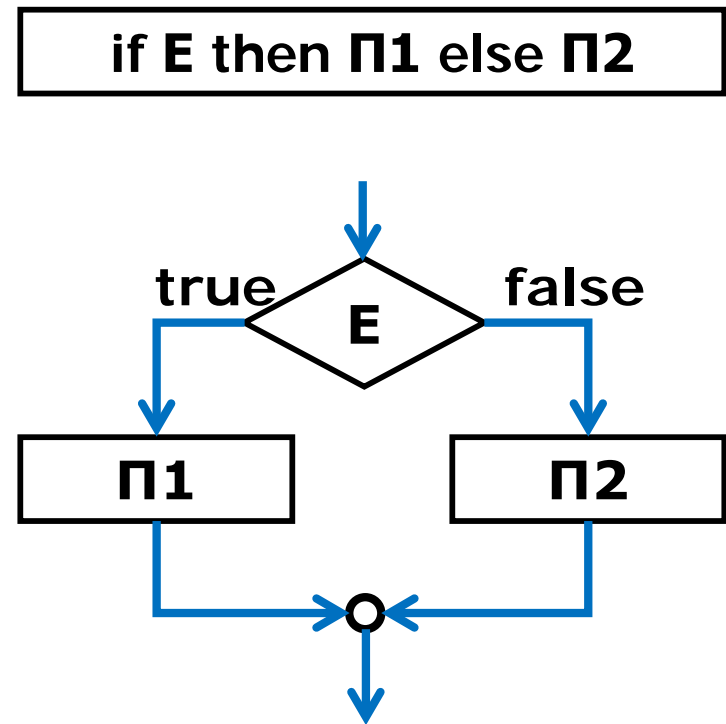
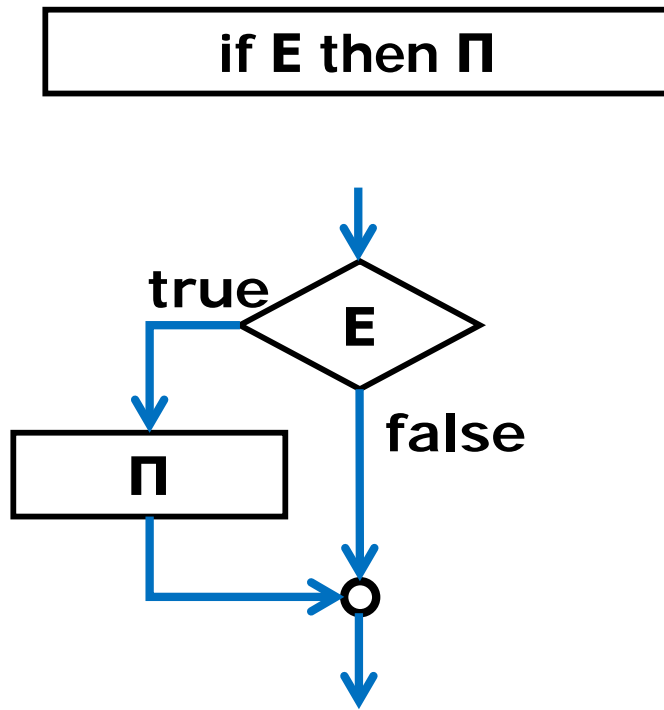
Προτάσεις διακλάδωσης υπό συνθήκη

Εντολές **if, if ... else**

Επιλογή - Διακλάδωση υπό συνθήκη

Η πρόταση **if** συναντάται σε όλες τις προστακτικές ΓΠ

Απλή μορφή



Επιλογή - Διακλάδωση υπό συνθήκη

Σύνθετη μορφή (φωλιασμένα if)

if E1 then Π1

else if E2 then Π2

else Π3

Γενική μορφή:

if E1 then Π1

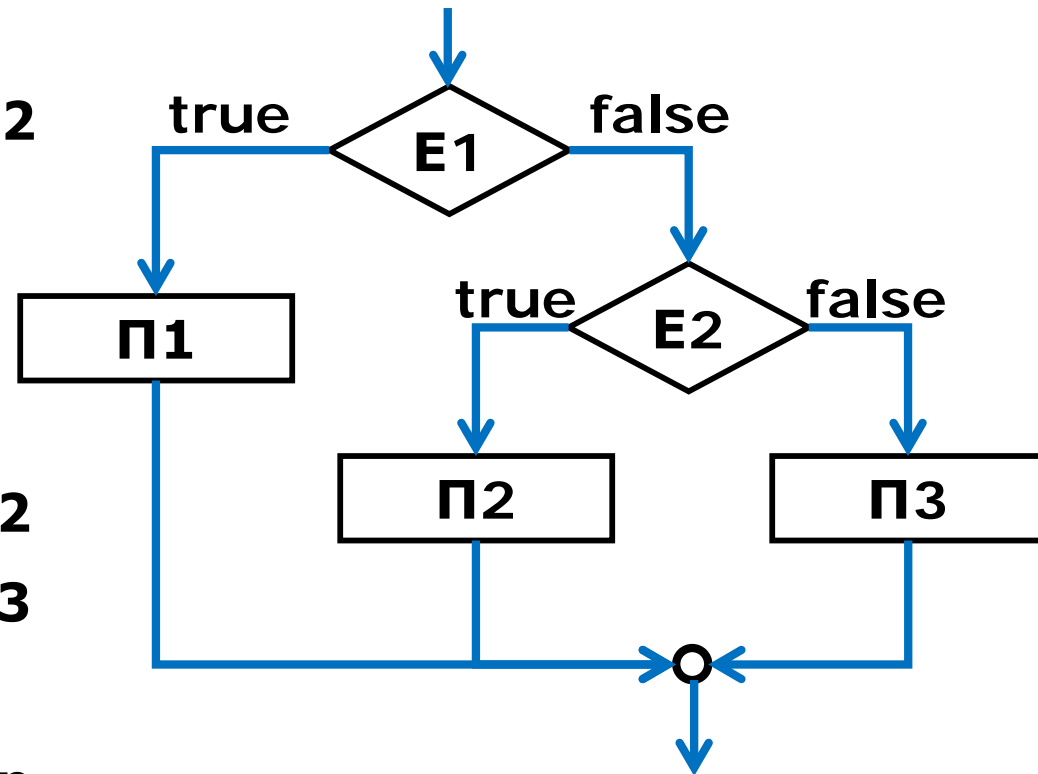
else if E2 then Π2

else if E3 then Π3

...

else if En then Πn

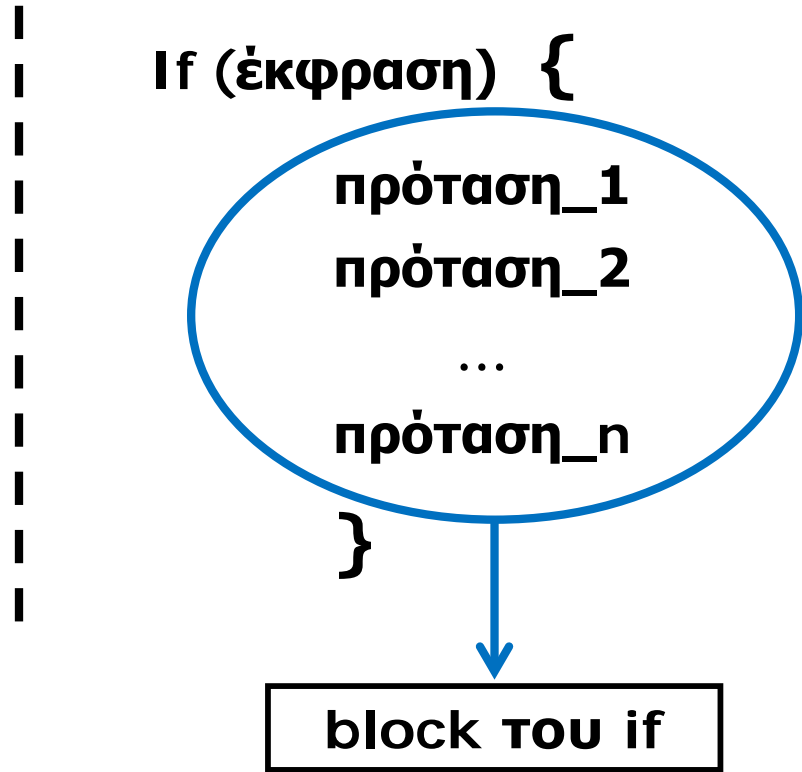
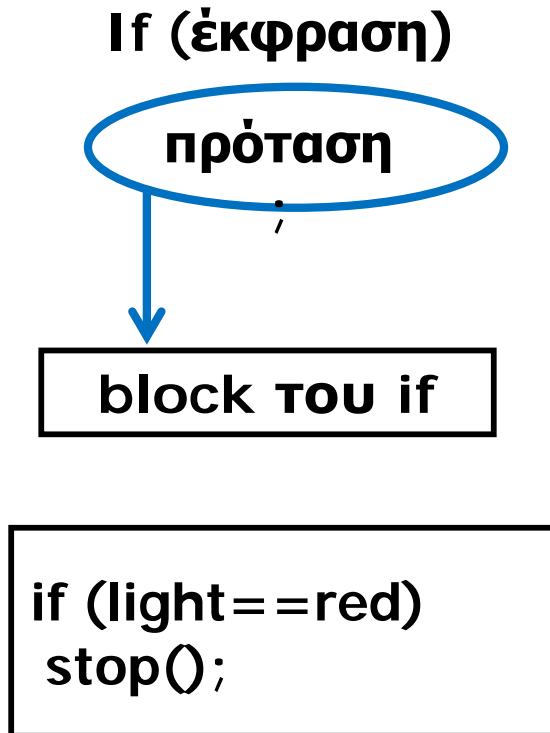
else Π0



Προτάσεις επιλογής στη C

Πρόταση **if**

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



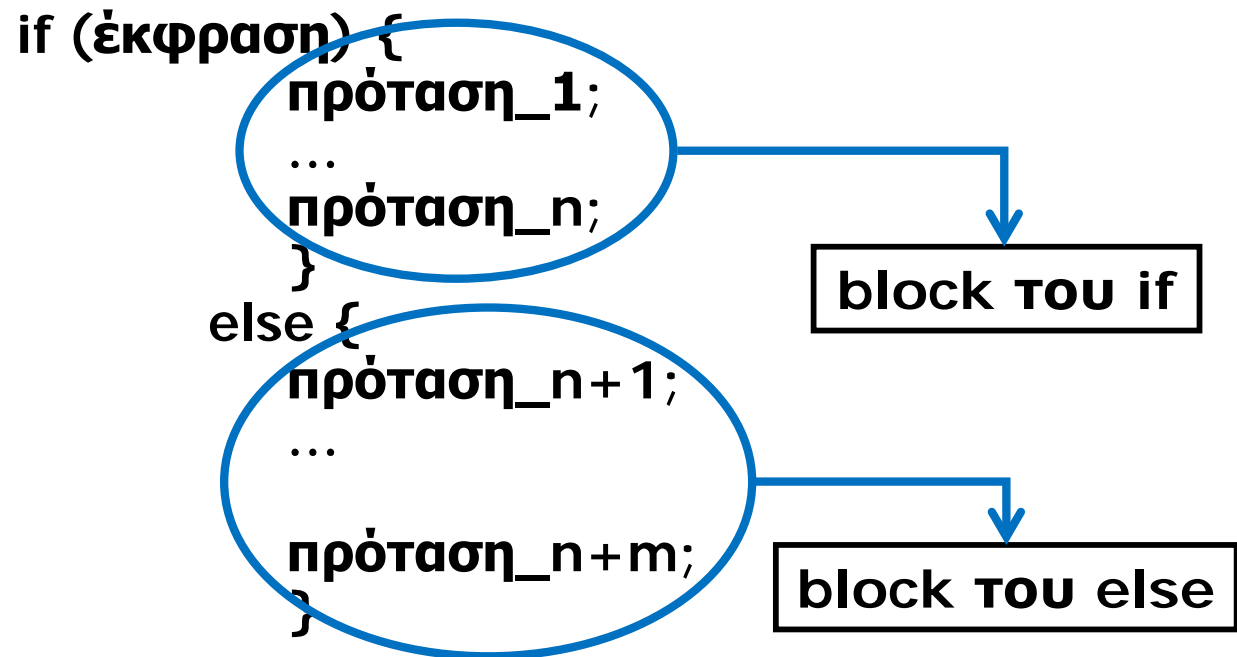
Προτάσεις επιλογής στη C

Πρόταση **if ... else**

Εκτελεί διαφορετικές προτάσεις ή ομάδες προτάσεων, ανάλογα με το αν μια έκφραση είναι αληθής ή ψευδής

```
if (έκφραση)
    πρόταση_1;
else
    προταση_2;
```

```
if (light == red)
    stop();
else
    go();
```



Προτάσεις επιλογής στη C

Ψευδοκώδικας

Αν ο βαθμός ενός φοιτητή είναι μεγαλύτερος του ή ίσος με 5
τύπωσε μήνυμα ότι προβιβαστικέ

αλλιώς

τύπωσε μήνυμα ότι απέτυχε

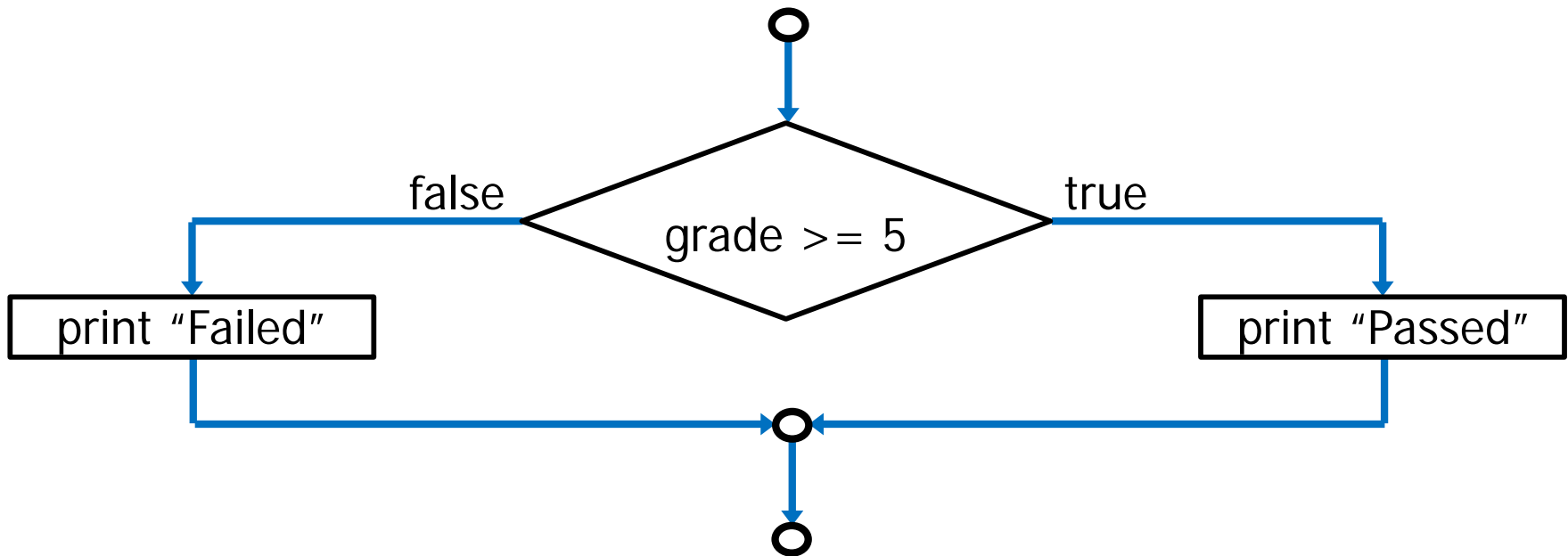
Κώδικας σε C:

```
if (grade >= 5)
    printf("Passed\n");
else
    printf("Failed\n");
```



Προτάσεις επιλογής στη C

Διάγραμμα ροής



Παράδειγμα 1

Δώστε τη δήλωση και τον ορισμό της συνάρτησης `min` για τον υπολογισμό του μικρότερου μεταξύ δύο ακεραίων αριθμών

- Παίρνει ως παραμέτρους δυο ακεραίους αριθμούς και επιστρέφει τον μικρότερο

```
int min(int a, int b)  
{  
    int minimum;  
    if (a < b)  
        minimum = a;  
    else  
        minimum = b;  
    return minimum;  
}
```



Παράδειγμα 1

Εναλλακτικές λύσεις

```
int min(int a, int b)
{ int minimum;
  if (a < b)
    minimum = a;
  else
    minimum = b;
  return minimum;
}
```

```
int min(int a, int b)
{
  if (a < b)
    return a;
  else
    return b;
}
```

```
int min(int a, int b)
{
  if (a < b)
    return a;
  return b;
}
```

```
int min(int a, int b)
{
  int minimum;
  minimum = b;
  if (a < b)
    minimum = a;
  return minimum;
}
```



Παράδειγμα 1

```
int min(int a, int b)
{ int minimum;
  if (a < b)
    minimum = a;
  else
    minimum = b;
  return minimum;
}
```

```
int min(int a, int b)
{
  return (a < b) ? a : b;
}
```

Υποθετικός τελεστής:
επιστρέφει **a** αν η συνθήκη
(**a < b**) είναι αληθής και **b** αν
είναι ψευδής



Παράδειγμα 2

Δώστε τη δήλωση και τον ορισμό της συνάρτησης **min3** για τον υπολογισμό του μικρότερου μεταξύ **τριών** ακεραίων αριθμών

Χρήση **nested if .. else**

```
int min3(int a, int b, int c)
{
    int minimum;
    if (a < b)
        if (a < c)
            minimum = a;
        else
            minimum = c;
    else
        if (b < c)
            minimum = b;
        else
            minimum = c;
    return minimum;
}
```



Παράδειγμα 2

Εναλλακτικές λύσεις

```
int min3(int a, int b, int c)
{
    int minimum;
    if (a < b)
        if (a < c)
            minimum = a;
        else
            minimum = c;
    else
        if (b < c)
            minimum = b;
        else
            minimum = c;
    return minimum;
}
```

```
int min3(int a, int b, int c)
{
    int minimum;
    if (a < b)
        minimum = a;
    else
        minimum = b;
    if (c < minimum)
        minimum = c;
    return minimum;
}
```

Μέρος 3^ο

Έκφραση συνθήκης

Έκφραση συνθήκης

Συσχετιστικοί και Λογικοί τελεστές

Περί εκφράσεων συνθήκης

Χρήση συσχετιστικών τελεστών (ή τελεστών σύγκρισης)

`<`, `>`, `<=`, `>=`, `==` (ελέγχου ισότητας), `!=` (διάφορο)

- Το αποτέλεσμα έκφρασης που περιέχει τέτοιο τελεστή είναι αληθές (true) ή ψευδές (false)
- Στη C, η τιμή είναι 0 για false και 1 για true
- Τύποι τελεστών (operands): int, char, float, double
- Παραδείγματα:

(i >= 8)

(x < y)

(f == 3.456)

('a' >= 'b')



Περί εκφράσεων συνθήκης

Χρήση λογικών τελεστών

&& (AND), **||** (OR), **!** (NOT)

- Συνένωση δύο ή περισσότερων εκφράσεων σύγκρισης
- Τύποι τελεστών (operands): int, char
- Παραδείγματα:

(a >= 0 && a <= 100)

(s < 10 || s > 100)

!(i == 5 && k == 10)



Περί εκφράσεων συνθήκης

&& (AND)

Ελέγχουμε αν ισχύουν όλες οι επιμέρους εκφράσεις

Πίνακας 1: Πίνακας αληθείας

| a | b | a&&b |
|----------|----------|---------------------|
| 1 | 1 | 1 |
| 1 | 0 | 0 |
| 0 | 1 | 0 |
| 0 | 0 | 0 |

|| (OR)

Ελέγχουμε αν ισχύει τουλάχιστο μία από τις επιμέρους εκφράσεις

Πίνακας 2: Πίνακας αληθείας

| a | b | a b |
|----------|----------|-------------|
| 1 | 1 | 1 |
| 1 | 0 | 1 |
| 0 | 1 | 1 |
| 0 | 0 | 0 |



Περί εκφράσεων συνθήκης

! (NOT)

Ελέγχουμε αν ισχύει το αντίθετο μιας έκφρασης

Πίνακας 3: Πίνακας αληθείας

| a | !a |
|----------|-----------|
| 1 | 0 |
| 0 | 1 |



Αποτίμηση λογικών εκφράσεων

Αρχίζει από τα αριστερά και προχωρεί μέχρι το σημείο που χρειάζεται

Παραδείγματα:

(0 && E) \rightarrow 0

(1 || E) \rightarrow 1

- η αποτίμηση της έκφρασης E δεν χρειάζεται σε καμία από τις παραπάνω περιπτώσεις



Αποτίμηση λογικών εκφράσεων

Προσοχή

- Σε μία σύζευξη **(E1 && E2)**, η συνθήκη E1 πρέπει να αποτελεί τον αριστερό τελεστή, εάν στην περίπτωση που αποτιμείται σε 0, τυχόν αποτίμηση της E2 οδηγεί σε πρόβλημα (με αυτή την σειρά αποτρέπεται η αποτίμηση της E2)
- Παράδειγμα:

$$((y / x) > 2 \ \&\& \ x \neq 0) \text{ vs. } (x \neq 0 \ \&\& \ (y / x) > 2)$$



Λογικές εκφράσεις σε C

Τα x και y είναι μεγαλύτερα του z

(x>z && y>z)

Το x είναι ίσο με 2.5 ή με 10.32

(x==2.5 || x==10.32)

Το a είναι μεταξύ του b και του c

(a>=b && a<=c)



Λογικές εκφράσεις σε C

Το x είναι μικρότερο του 0 ή μεταξύ του 10 και του 1000

```
(x < 0) || (x >= 10 && x <= 1000)
```

Το k είναι αγγλικός χαρακτήρας

```
(k >= 'a' && k <= 'z') || (k >= 'A' && k <= 'Z')
```



Μέρος 4^ο

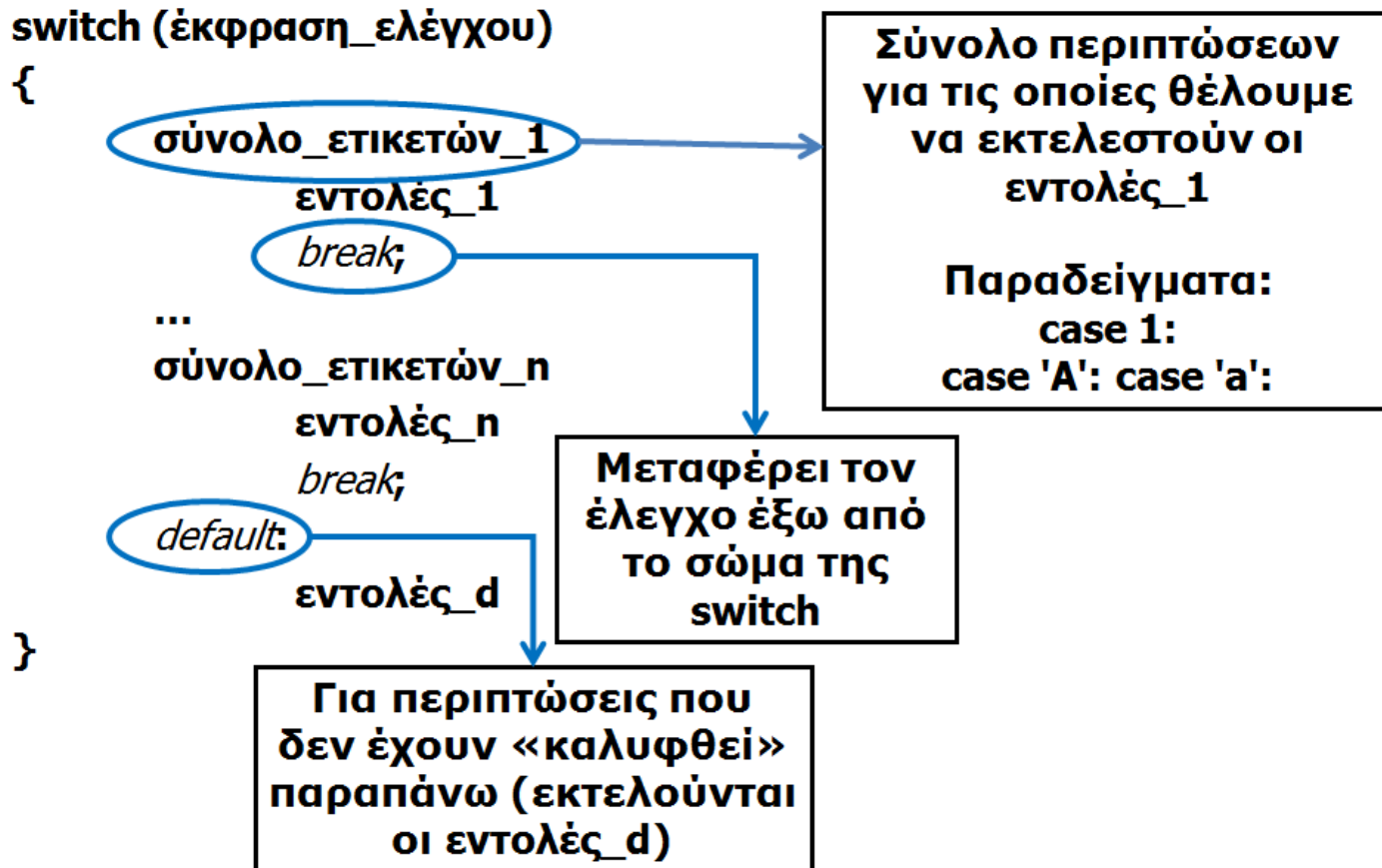
Οι προτάσεις **switch** και **break**

Οι προτάσεις **switch** και **break**

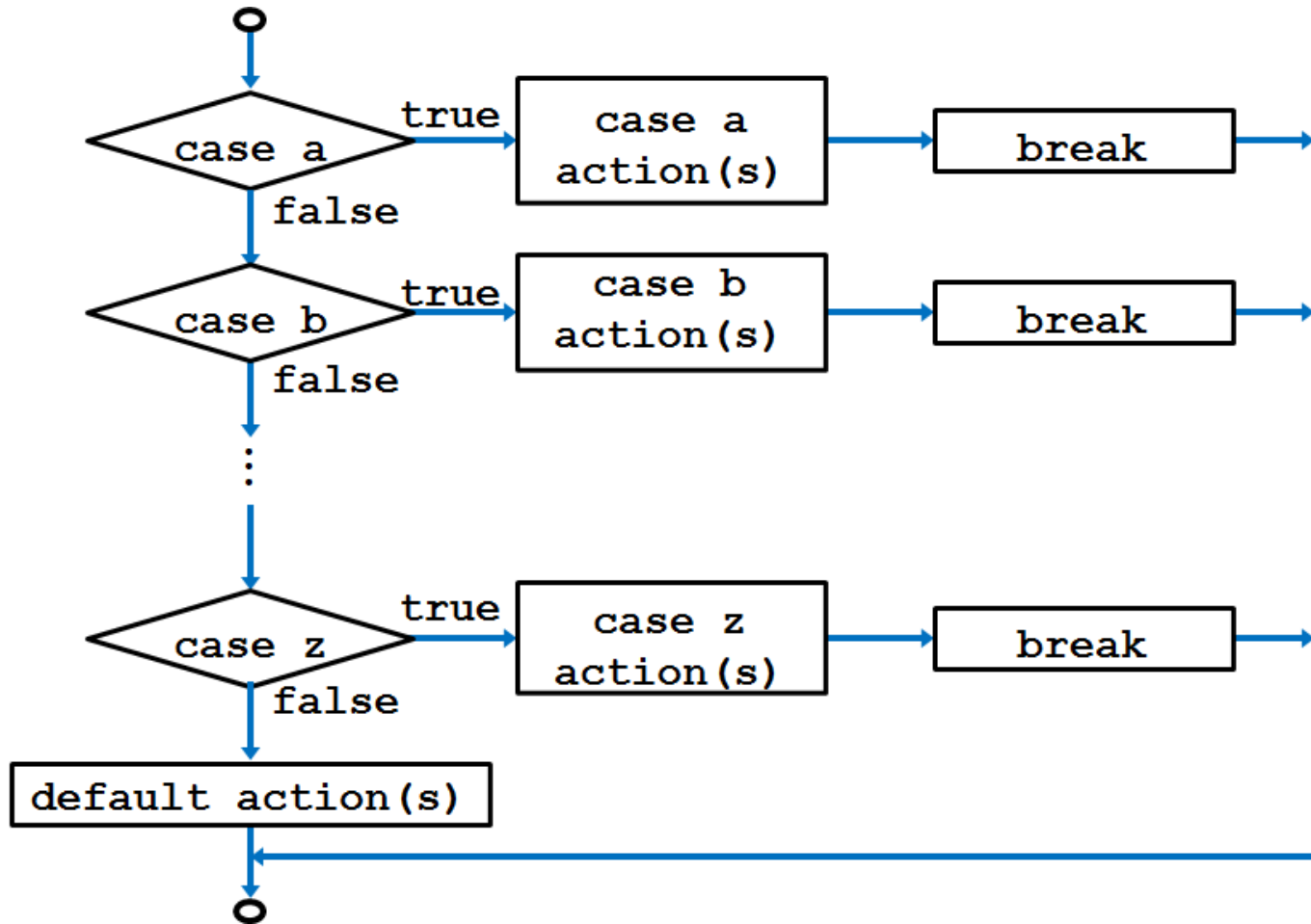
Συσχετιστικοί και Λογικοί τελεστές

Η πρόταση switch

Γενική μορφή



Η πρόταση switch



Η πρόταση switch

Προσοχή

Η **έκφραση_ελέγχου** πρέπει να είναι βαθμωτού τύπου, π.χ. char ή int, αλλά όχι float ή double

- Κάθε **case** πρέπει να έχει μια **int** ή **char** σταθερά ή σταθερά έκφραση

Δύο **case** δεν μπορούν να έχουν την ίδια τιμή



Η πρόταση switch

Οι προτάσεις κάτω από την ετικέτα **default** εκτελούνται όταν καμιά από τις **case** ετικέτες δεν ικανοποιείται

Η **default** δεν είναι απαραίτητα η τελευταία ετικέτα

Σημασία και χρήση της **break**

- Ροή προγράμματος (βλέπε προηγούμενη διαφάνεια)



Η πρόταση switch – Παράδειγμα

```
char letter_grade;
int count_a, count_b,
count_other;
count_a = count_b =
count_other = 0;
...
switch(letter_grade) {
    case 'A':
        counta++;
        message_excellent();
        break;
    case 'B':
        countb++;
        message_very_good();
        break;
    case 'C':
    case 'D':
        count_other++;
        message_other();
        break;
    default:
        message_error();
}
```



Λάθη που γίνονται συχνά

Χρήση τελεστή ανάθεσης (=) αντί του τελεστή ελέγχου ισότητας (==)

if (x = 10) αντί του ορθού **if (x == 10)**

Παράλειψη παρένθεσης στη συνθήκη

if x > 4 αντί του ορθού **if (x > 4)**

Κακή μετατροπή μαθηματικής έκφρασης σε έκφραση της C

(0 <= x <= 4) αντί του ορθού **(0 <= x && x <= 4)**



Λάθη που γίνονται συχνά

Παράλειψη αγκυλών στο σώμα του **if** ή του **else**

```
if (x > 0)
    sum = sum + x;
    printf("Greater than
0");
else
    printf("Less than 0");
```

*αντί του
ορθού*

```
if (x > 0){
    sum = sum + x;
    printf("Greater than 0");
}
else
    printf("Less than 0");
```



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 Επιλογής». 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

