

ΕΡΓΑΣΙΑ 2

ΔΟΜΕΣ ΕΛΕΓΧΟΥ

Χρήστος Μαργιώλης – Τμήμα 9

Τί γνωρίζετε για τις εντολές if και switch;

Οι εντολές `if` και `switch` χρησιμοποιούνται για τον έλεγχο παραστάσεων, ώστε να εκτελεστούν και οι ανάλογες εντολές που έχει ορίσει ο προγραμματιστής να εκτελεστούν ανάλογα με το ποια τιμή δίνει η παράσταση. Οι τιμές που μπορεί να δώσει μια παράσταση είναι είτε 1 είτε 0, δηλαδή αληθής ή ψευδής αντίστοιχα.

Η `if` ελέγχει αν μια παράσταση είναι αληθής, και εκτελεί το τμήμα εντολών που υπάρχει κάτω από αυτήν – σε περίπτωση που δεν είναι αληθής, τότε δεν θα εκτελεστεί το τμήμα εντολών κάτω από την `if`, αλλά εάν υπάρχει και η εντολή `else` ή `else if`, θα εκτελεστούν αυτές με τον ίδιο τρόπο αντίστοιχα, ελέγχοντας άλλες παραστάσεις. Αν σε καμία περίπτωση δεν είναι αληθής κάποια από τις παραστάσεις, δεν εκτελείται κανένα τμήμα εντολών κάτω από τις εντολές `if`, `else`, ή `else-if` που έδωσαν ψευδείς τιμές.

Η `switch` ελέγχει αν η τιμή της παράστασης που της δίνεται, αντιστοιχίζεται με κάποιο από τα `cases` (περιπτώσεις) και εκτελεί τις κατάλληλες εντολές. Τα `cases` μπορούν να είναι μόνο ακέραιοι αριθμοί είτε μονοί χαρακτήρες. Ο έλεγχος γίνεται από πάνω προς τα κάτω και αν η τιμή της παράστασης ταυτίζεται με κάποιο από τα `cases`, τότε εκτελούνται όλες οι εντολές από εκεί και κάτω, εκτός και αν παρεμβεί κάποια εντολή `break`, η οποία θα σταματήσει την εκτέλεση εντολών μέσα στην `switch`. Εάν η τιμή τελικά δεν ταυτιστεί με κανένα `case`, είτε υπάρχει η εντολή `default` για να εκτελέσει κάποιες εντολές στην περίπτωση που δεν υπάρχει ταύτιση τιμής παράστασης με `case`, είτε δεν εκτελείται καμία εντολή μέσα στην `switch`. Η `default` μπορεί να γραφτεί σε όποια θέση θέλει ο προγραμματιστής, αν και συνηθίζεται να γράφεται στο τέλος, μετά από όλα τα `cases`

if.c

```
#include <stdio.h>

int main (int NoP, char **Params)
{
    int a, b, c;
    int Max;
    printf ("ΕΥΡΕΣΗ ΜΕΓΙΣΤΟΥ ΑΡΙΘΜΟΥ\n\n");
    printf ("Εισάγετε τον πρώτο αριθμό   : ");
    scanf ("%d", &a);
    printf ("Εισάγετε τον δεύτερο αριθμό : ");
    scanf ("%d", &b);
    printf ("Εισάγετε τον τρίτο αριθμό   : ");
    scanf ("%d", &c);
    Max = a;
    if (b > Max)
        Max = b;
    Max = (c > Max)? c: Max;
    printf ("Ο μεγαλύτερος είναι ο %d και εισήχθη:\n", Max);
    if (Max == a)
        printf("  1ος\n");
    if (Max == b)
        printf("  2ος\n");
    if (Max == c)
        printf("  3ος\n");
    return 124;
}
```

if.c – Μελέτη λειτουργίας

Το πρόγραμμα έχει ως στόχο την εύρεση του μέγιστου από τους 3 ακέραιους αριθμούς που δίνονται από τον χρήστη. Αρχικά εισάγονται από το πληκτρολόγιο οι 3 αριθμοί και στην συνέχεια το πρόγραμμα βρίσκει τον μεγαλύτερο. Αφού βρεθεί, ελέγχει να δει με ποιον από τους 3 αριθμούς αντιστοιχίζεται ο μέγιστος, ώστε να εμφανίσει το πρόγραμμα στην οθόνη και την σειρά με την οποία εισήχθη ο αριθμός.

Ερώτημα 2 (πρώτο μέρος)

if1.c

```
#include <stdio.h>

int main (int NoP, char **Params)
{
    int a, b, c;
    int Max;
    printf ("ΕΥΡΕΣΗ ΜΕΓΙΣΤΟΥ ΑΡΙΘΜΟΥ\n\n");
    printf ("Εισάγετε τον πρώτο αριθμό   : ");
    scanf ("%d", &a);
    printf ("Εισάγετε τον δεύτερο αριθμό : ");
    scanf ("%d", &b);
    printf ("Εισάγετε τον τρίτο αριθμό   : ");
    scanf ("%d", &c);
    if (a > b)
        if (a > c)
            printf("Ο μεγαλύτερος είναι ο %d και εισήχθη
1ος\n", a);
        else
            if (a == c)
                printf ("Ο μεγαλύτερος είναι ο %d και εισήχθη
1ος και 3ος\n", a);
            else
                printf ("Ο μεγαλύτερος είναι ο %d και εισήχθη
3ος\n", c);
        else
            if (a == b)
                if (a == c)
                    printf ("Ο μεγαλύτερος είναι ο %d και εισήχθη
1ος, 2ος και 3ος\n", a);
                else
                    if (a > c)
                        printf ("Ο μεγαλύτερος είναι ο %d και
εισήχθη 1ος και 2ος\n", a);
                    else
                        printf ("Ο μεγαλύτερος είναι ο %d και
εισήχθη 3ος\n", c);
                else
                    if (b > c)
                        printf ("Ο μεγαλύτερος είναι ο %d και εισήχθη
2ος\n", b);
                    else
                        if (b == c)
                            printf ("Ο μεγαλύτερος είναι ο %d και
εισήχθη 2ος και 3ος\n", a);
                        else
```

```

        printf ("Ο μεγαλύτερος είναι ο %d και
εισήχθη 3ος\n", c);
        return 32;
}

```

if1.c – Μελέτη λειτουργίας

Το πρόγραμμα αυτό, όπως και το `if.c`, βρίσκει τον μέγιστο ακέραιο αριθμό που δίνεται από το πληκτρολόγιο, καθώς και την σειρά με την οποία εισήχθη. Η διαφορά όμως αυτού του προγράμματος με το προηγούμενο είναι ότι δεν χρησιμοποιείται κάποια επιπλέον μεταβλητή `Max` για να βρεθεί ο μέγιστος αριθμός, αλλά μέσω *nested if-else* εντολών, το πρόγραμμα ελέγχει όλους τις πιθανους συνδυασμούς συγκρίσεων αριθμών, και έτσι, *αν και με περισσότερες εντολές και μεγαλύτερη πολυπλοκότητα*, βρίσκεται και ο μέγιστος αριθμός και η σειρά εισαγωγής του.

Ερώτημα 2 (δεύτερο μέρος)

switch.c

```

#include <stdio.h>
#define t 6

int main ()
{
    int i = 7;
    switch (i)
    {
        case 3: printf ("3\n");
        case 5: printf ("5\n"); break;
        case 3 + 4: printf ("7\n");
        default: printf ("default\n");
        case 1: printf ("1\n"); break;
        case t: printf ("t\n");
    }
}

```

switch.c – Μελέτη λειτουργίας

Το πρόγραμμα έχει ως στόχο την εύρεση του αριθμού `i`. Προκειμένου να πραγματοποιηθεί αυτή η λειτουργία, χρησιμοποιείται η εντολή `switch`, η οποία ελέγχει αν ο αριθμός `i` αντιστοιχίζεται με κάποιον από τις τιμές των `cases` που υπάρχουν μέσα στην εντολή. Σε περίπτωση που τελικά ο `i` αντιστοιχίζεται με κάποιον από τους αριθμούς της `switch`, εμφανίζεται στην οθόνη ποιός ήταν αυτός ο αριθμός. Στην προκειμένη περίπτωση, η τιμή του `i` (7) έχει δοθεί ήδη μέσα στο πρόγραμμα, οπότε στην οθόνη θα εμφανιστεί "7", εφόσον το `case 3 + 4` ελέγχει αν ο αριθμός είναι 7.

Ερώτημα 2 (τρίτο μέρος)

quadratic-equation.c – Άσκηση 3

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

void linearEquation();
void quadraticEquation();

int main(int argc, char **argv)
{
    double a, b, c;

    printf("a: ");
    scanf("%lf", &a);
    printf("b: ");
    scanf("%lf", &b);
    printf("c: ");
    scanf("%lf", &c);

    if (a != 0)
        quadraticEquation(a, b, c);
    else
        linearEquation(b, c);

    return 0;
}
```

```

void linearEquation(double b, double c)
{
    double x;

    if (b != 0)
    {
        x = -c / b;
        printf("x = %.21f\n", x);
    }
    else
        printf("Infinite solutions.\n");
}

void quadraticEquation(double a, double b, double c)
{
    double x, x1, x2, D = pow(b, 2) - 4*(a*c);

    if (D > 0)
    {
        x1 = (-b + sqrt(D)) / (2*a);
        x2 = (-b - sqrt(D)) / (2*a);
        printf("x1 = %.21f\nx2 = %.21f\n", x1, x2);
    }
    else if (D == 0)
    {
        x = (-b) / (2*a);
        printf("x = %.21f\n", x);
    }
    else
    {
        printf("There are no solutions.\n");
    }
}

```

Ερώτημα 3 (πρώτο μέρος)

Περιγραφή υλοποίησης άσκησης 3

- **Μεταβλητές:**

a: Πρώτος όρος ($ax^2 + bx + c$)

b: Δεύτερος όρος ($ax^2 + bx + c$)

c: Τρίτος όρος ($ax^2 + bx + c$)

x: Λύση εξίσωσης όταν υπάρχει 1 ρίζα

x1: Λύση 1 όταν υπάρχουν 2 ρίζες

x2: Λύση 2 όταν υπάρχουν 2 ρίζες

D: Διακρίνουσα

- **Λειτουργία του προγράμματος:**

Αρχικά διάβαζει τους 3 όρους της δευτεροβάθμιας εξίσωσης και στην συνέχεια ελέγχει αν $a \neq 0$ ώστε να κρίνει αν η εξίσωση είναι δευτεροβάθμια ή αν τελικά είναι πρωτοβάθμια – αν $a = 0$, τότε καλεί την συνάρτηση για την λύση πρωτοβάθμιας εξίσωσης, και αν $a \neq 0$, καλεί την συνάρτηση για την λύση δευτεροβάθμιας εξίσωσης.

Στην συνάρτηση πρωτοβάθμιας εξίσωσης, το πρόγραμμα ελέγχει αν $b \neq 0$, ώστε να προχωρήσει και να βρει το x, ή να εμφανίσει στην οθόνη ότι υπάρχουν άπειρες ρίζες.

Στην συνάρτηση δευτεροβάθμιας εξίσωσης, το πρόγραμμα υπολογίζει την διακρίνουσα, και εξετάζει τις 3 περιπτώσεις που μπορεί να προκύψουν ($D > 0$, $D = 0$ και $D < 0$), και στην συνέχεια κάνει τις κατάλληλες πράξεις.

Ερώτημα 3 (δεύτερο μέρος)

int-comparison.c – Άσκηση 4

```
#include <stdio.h>

int main(int argc, char **argv)
{
    int intArr[3], maxInt, index;

    printf("Integer 1: ");
    scanf("%d", &intArr[0]);

    printf("Integer 2: ");
    scanf("%d", &intArr[1]);

    printf("Integer 3: ");
    scanf("%d", &intArr[2]);

    maxInt = (intArr[0] > intArr[1]) ? ((intArr[0] >
intArr[2]) ? intArr[0] : intArr[2]) : ((intArr[1] >
intArr[2]) ? intArr[1] : intArr[2]);

    pos = (maxInt == intArr[0]) ? 1 : ((maxInt ==
intArr[1]) ? 2 : (maxInt == intArr[1]) ? 3 : 4);

    printf("The biggest number is %d in position %d.\n",
maxInt, index);

    return 0;
}
```

Ερώτημα 4 (πρώτο μέρος)

Περιγραφή υλοποίησης άσκησης 4

- **Μεταβλητές:**

`intArr[3]`: Array που καταχωρούνται οι 3 ακέραιοι

`maxInt`: Μεγαλύτερος ακέραιος

`index`: Θέση μεγαλύτερο ακεραίου στον πίνακα / σειρά εισαγωγής

- **Λειτουργία του προγράμματος:**

Αρχικά διαβάζει και καταχωρεί στον πίνακα τους 3 ακέραιους, και στην συνέχεια συγκρίνει ποιος από τους 3 είναι ο μεγαλύτερος. Συγκρίνει τον 1ο με τον 2ο ακέραιο, και αν ο 1ος είναι μεγαλύτερος, τότε τον συγκρίνει με τον 3ο ακέραιο – αν ο 1ος παραμένει μεγαλύτερος, καταχωρεί την τιμή του στην μεταβλητή `maxInt`, αλλιώς της καταχωρεί την μεταβλητή του 3ου ακεραίου. Αν στην πρώτη σύγκριση (`intArr[0] > intArr[1]`) ο 2ος ακέραιος είναι ο μεγαλύτερος, τότε τον συγκρίνει με τον 3ο ακέραιο, και αν ο 2ος παραμένει μεγαλύτερος, καταχωρεί την τιμή του στην `maxInt`, αλλιώς της καταχωρεί την τιμή του 3ου. Τέλος εμφανίζει τον μεγαλύτερο ακέραιο και την σειρά με την οποία εισήχθη.

Ερώτημα 4 (δεύτερο μέρος)

Πηγές

- C Programming Language – Brian W. Kernighan & Dennis M. Ritchie
- [The if and switch statement](#)

Για την δημιουργία τις εργασίας χρησιμοποιήθηκαν

- **Editors:** Visual Studio Code
- **Compiler:** gcc
- **Συγγραφή:** LibreOffice Writer
- **Γραμματοσειρές:** Liberation Sans για τα κείμενα και Liberation Mono για τους κώδικες
- **Λειτουργικό σύστημα:** Linux Mint Cinnamon 19.2