

Laborator 7. Instrucțiuni de decizie: *if*, *switch*. Exemple de programe. Program temă individuală 2

7.1. Instrucțiunea *if*

Instrucțiunea *if* permite impunerea unei condiții în interiorul unei funcții, în vederea execuției condiționate a unei instrucțiuni (simple sau compuse).

Sintaxa:

```
if (expresie)
    instructiune1
[else
    instructiune2]
```

Ex1. Se dă funcția: $f(x,y) = \begin{cases} \frac{x^2+y}{x-3}, & \text{daca } x \leq 4 \\ 2x+y, & \text{daca } 4 < x \leq 10 \\ xy^2, & \text{daca } x > 10 \end{cases}$. Să se calculeze și să se afișeze valoarea funcției în trei

puncte distincte din domeniul de definiție, corespunzătoare celor trei ramuri ale funcției. Afișarea se va face într-un tabel de forma:

Nr.	x	y	f(x,y)
1.			
2.			
3.			

```
#include<stdio.h>
#include<conio.h>
void main(void)
{
    double x1, y1, x2, y2, x3, y3, f1, f2, f3;
    printf("Primul punct:\nx = ");
    scanf("%lf", &x1);
    printf("y = ");
    scanf("%lf", &y1);
    if (x1 <= 4)
        f1 = (x1 * x1 + y1) / (x1 - 3);
    else if (x1 > 10)
        f1 = x1 * y1 * y1;
    else
        f1 = 2 * x1 + y1;
    printf("Al doilea punct:\nx = ");
    scanf("%lf", &x2);
    printf("y = ");
    scanf("%lf", &y2);
    if (x2 <= 4)
        f2 = (x2 * x2 + y2) / (x2 - 3);
    else if (x2 > 10)
        f2 = x2 * y2 * y2;
    else
        f2 = 2 * x2 + y2;
    printf("Al treilea punct:\nx = ");
    scanf("%lf", &x3);
    printf("y = ");
    scanf("%lf", &y3);
```

```

if (x3 <= 4)
    f3 = (x3 * x3 + y3) / (x3 - 3);
else if (x3 > 10)
    f3 = x3 * y3 * y3;
else
    f3 = 2 * x3 + y3;
printf("=====\n");
printf("| Nr. |      x      |      y      |      f(x,y)      |\n");
printf("-----\n");
printf("| 1. |%11.4lf|%11.4lf|%15.4lf|\n", x1, y1, f1);
printf("-----\n");
printf("| 2. |%11.4lf|%11.4lf|%15.4lf|\n", x2, y2, f2);
printf("-----\n");
printf("| 3. |%11.4lf|%11.4lf|%15.4lf|\n", x3, y3, f3);
printf("=====\n");
getch();
}

```

7.2 Instrucțiunea switch

Instrucțiunea *switch* permite selectarea unui caz dintr-o mulțime finită de variante și execuția condiționată a unei secvențe de instrucțiuni, în funcție de încadrarea într-o ramură.

Sintaxa:

```

switch
case (expresie)
{
    case c1: secventa instructiunii1 [break;]
    case c2: secventa instructiunii2 [break;]
    ...
    case c_n: secventa instructiunii_n [break;]
    [default: secventa instructiunii_n_plus_1]
}

```

Instrucțiunea *break* realizează saltul la sfârșitul instrucțiunii *switch*, iar în absența ei se vor executa și următoarele secvențe de instrucțiuni.

Ex2. Să se scrie un program C, care să citească două numere întregi fără semn *x* și *y* de la tastatură și un operator logic pe biți *op* (&, ^, |). Să se calculeze rezultatul operației *x op y* și să se afișeze rezultatul.

```

#include<stdio.h>
#include<conio.h>
#include<string.h>
void main(void)
{
    unsigned x, y, val, j = 1;
    char op, mesaj[20], mesaj_and[] = "si logic pe biti.";
    char mesaj_xor[] = "sau exclusiv pe biti.", mesaj_or[] = "sau logic pe biti.";
    printf("x = ");
    scanf("%u", &x);
    printf("y = ");
    scanf("%u", &y);
    printf("Introduceti operatorul logic pe biti: ");
    op = getch();
    switch (op)
    {
        case '&': val = x & y; strcpy(mesaj, mesaj_and); break;
        case '^': val = x ^ y; strcpy(mesaj, mesaj_xor); break;
        case '|': val = x | y; strcpy(mesaj, mesaj_or); break;
        default: printf("\nOperator necunoscut!!"); j=0;
    }
    if (j!=0)
        printf("\nOperatia este: %s\nRezultatul operatiei este %u.\n", mesaj, val);
    getch();
}

```

5.12 Exerciții propuse spre rezolvare

1. Să se scrie un program în care să se verifice, utilizând instrucțiunea if, dacă un an este bisect sau nu. Regula de stabilire dacă un an este bisect este următoarea: "Un an este bisect dacă este divizibil cu 4, exceptând cazurile când este divizibil cu 100 fără a fi divizibil cu 400" (http://ro.wikipedia.org/wiki/An_bisect).

Câteva exemple pentru a înțelege mai bine această regulă:

- ani biseți "obișnuiți": 1872, 1912, 1960, 1996, 2008, 2012, 2016, 2020, 2024, 2028, 2032, 2036, 2040...
- excepții de la regula de bază (nu sunt biseți): 1800, 1900, 2100, 2200...
- sunt totuși ani biseți: 2000, 2400, 2800 ...

2. Să se scrie un program care citește o cifră de la 1 la 7, reprezentând o zi din săptămână. Să se scrie un program C care afișează numele zilei respective (luni, marți etc.). În cazul în care cifra citită nu este în intervalul 1..7, să se afișeze un mesaj de eroare.