

## Laborator 4

### 1. Instrucțiunea `while`

Instrucțiunea `while` este o instrucțiune repetitivă (de ciclare). Forma generală a instrucțiunii este următoarea:

```
while (expresie) instrucțiune
```

#### Mod de execuție:

1. Se evaluează *expresie*.
2. Dacă valoarea este diferită de zero (condiția este îndeplinită) se execută *instrucțiune* și se reia pasul 1, altfel se trece la instrucțiunea următoare instrucțiunii *while*.

Altfel spus, se execută în mod repetat *instrucțiune* cât timp valoarea expresiei din paranteză este diferită de zero.

Dacă expresia are valoarea zero (condiția este falsă) la prima evaluare, atunci instrucțiune nu se execută niciodată.

Trebuie să se aibă în vedere existența în *instrucțiune* a cel puțin unei prelucrări care să asigure încheierea execuției repetitive, altfel programul se blochează într-o buclă infinită. De asemenea, o parte din datele prelucrate în corpul ciclului trebuie inițializate înainte de a se ajunge la execuția instrucțiunii de ciclare.

#### Exemplu

```
x = 1;
while (x<=4)
{
    printf ("x = %d", x) ;
    x = x+1;
}
```

La execuția acestei secvențe de program se va afișa:

```
x = 1
x = 2
x = 3
x = 4
```

## 2.Probleme

1. Să se scrie un program în C care afișează suma primelor  $n$  numere naturale
2. Se citesc numere întregi până la întâlnirea lui zero. Să se calculeze suma numerelor citite.
3. Se citesc numere întregi până la întâlnirea lui zero. Să se afișeze suma numerelor pare și produsul numerelor impare.
4. Se citesc numere întregi până la întâlnirea lui zero. Să se afișeze câte sunt pozitive și câte sunt negative.
5. Se citește un număr natural  $n$ . Să se afișeze cifrele numărului.
6. Se citește un număr natural  $n$ . Să se afișeze dacă numărul este prim.
7. Se citesc două numere naturale  $a$  și  $b$ . Să se afișeze cel mai mare divizor comun al celor două numere.

## 3.Instrucțiunea do-while

Forma generală a instrucțiunii este următoarea:

```
do
    instrucțiune
while (expresie);
```

### Mod de execuție:

1. Se execută instrucțiune.
2. Se evaluează expresia din pranteză.
3. Dacă valoarea expresiei este diferită de zero se reia pasul 1, altfel se trece la instrucțiunea aflată după do-while.

Altfel spus, instrucțiune se execută repetat cât timp valoarea expresiei din paranteză este diferită de zero.

Dacă expresia are valoarea zero (condiția este falsă) de la început (prima evaluare), atunci instrucțiune se execută o dată. Cu alte cuvinte, în cazul instrucțiunii do-while instrucțiune se execută cel puțin o dată.

Ca și la instrucțiunea while, instrucțiune trebuie să conțină cel puțin o prelucrare care să asigure încheierea execuției repetitive, altfel programul se va bloca într-o buclă infinită. De asemenea, o parte din datele prelucrate în corpul ciclului trebuie inițializate înainte de a se ajunge la execuția instrucțiunii de ciclare.

### Exemplu

```
x = 1 ;  
do {  
    printf ("x = %d", x) ;  
    x = x+1;  
} while (x<=4) ;
```

La execuția acestei secvențe de program se va afișa:

```
x = 1  
x = 2  
x = 3  
x = 4
```

### 4.Probleme

8. Să se scrie un program în C care afișează suma primelor  $n$  numere naturale.
9. Să se scrie un program ce calculează perimetrul și aria unui pătrat cu latura dată. Dacă valoarea introdusă de la tastatură nu poate reprezenta latura unui pătrat, se solicită introducerea sa din nou. Altfel spus, se va repeta operația de citire a valorii laturii cât timp această valoare este negativă.
10. Se citesc numere întregi până la întâlnirea lui zero. Să se calculeze suma numerelor citite.
11. Se citesc de la tastatură  $n$  numere întregi. Să se calculeze suma lor. Valoarea lui  $n$  trebuie să fie strict pozitivă. În caz contrar, programul va solicita din nou introducerea acestei valori.
12. Se citesc numere întregi până la întâlnirea lui zero. Să se afișeze media aritmetică a numerelor din intervalul  $[3, 10]$ .

### 5.Probleme suplimentare

1. Se citesc de la tastatură  $n$  numere întregi. Să se afișeze suma numerelor pozitive. Valoarea lui  $n$  trebuie să fie în intervalul  $[1,50]$ . În caz contrar, programul va solicita din nou introducerea acestei valori.
2. Se citesc numere până la întâlnirea lui 0. Să se afișeze numărul de elemente din intervalul  $[a, b]$ . Valorile lui  $a$  și  $b$  se vor citi de la tastatură.
3. Se citesc numere întregi până la întâlnirea lui zero. Să se afișeze câte sunt divizibile cu 3 și câte sunt mai mari decât 7.

4. Să se calculeze perimetrul și aria unui dreptunghi cu laturile date. Dacă una dintre valorile introduse de la tastatură nu poate reprezenta o latură a unui dreptunghi, se solicită introducerea sa din nou.
5. Se citește un număr natural  $n$ . Să se afișeze dacă  $n$  este număr perfect.
6. Se citește un număr natural  $n$ . Să se afișeze dacă  $n$  este palindrom.
7. Să se scrie un program care citește de la tastatură numere până la întâlnirea lui 0 și afișează media aritmetică a câte două numere citite (primul cu al doilea, al doilea cu al treilea, al treilea cu al patrulea, etc).
8. Să se scrie un program care citește de la tastatură numere până la întâlnirea lui 0 și afișează suma primului număr cu al treilea, al cincilea, etc. și produsul dintre al doilea număr, al patrulea, al șaselea, etc.