

Algoritmi. Implementarea algoritmilor în C++.**Structura repetitivă cu număr cunoscut de pași. Instrucțiunea *for*. Aplicații.**

Atunci când anumite operații trebuie repetate de un număr de ori cunoscut (de obicei de un număr mare de ori, care nu permite scrierea repetată a operațiilor în algoritm), se utilizează structura repetitivă *pentru* (în C++, instrucțiunea *for*).

De exemplu, pentru afișarea numerelor cuprinse între a și b , date ($a < b$) care au cifra unităților egală cu 3, este necesară **parcurea** tuturor numerelor mai mari egale cu a și mai mici egale cu b și verificarea cifrei unităților. Dacă $a=10$ și $b=1000$, ar trebui verificate $1000-9=991$ numere!!

Structura pentru / Instrucțiunea for

Pentru (*contor = val_initala; contor <= val_finala ; contor = contor + pas*)

Instructiuni

Sfarsit pentru

Efect:

Etapa 1. Se inițializează contorul (variabila care numără pașii)

Etapa 2. Se verifică *daca valoarea contorului este mai mica sau egala cu valoarea finala*

- Dacă DA, se execută *instructiuni*, apoi *contorul creste cu valoarea pasului* și revenim la Etapa 2
- Dacă NU este îndeplinită, se oprește execuția structurii *Pentru*

În C++

for (*contor = val_initala; contor <= val_finala; contor = contor + pas*)

Instructiuni

Obs: *daca sunt două sau mai multe instrucțiuni în for se vor grupa între acolade {...}*

<i>Exemplul 1: Afișarea primelor n numere naturale</i>	
Algoritm	Program C++
Date de intrare: n Date de iesire: numerele cuprinse între 1 și n Citeste n Pentru (i=1; i<=n; i=i+1) scrie i sf. Pentru	<pre>#include <iostream> using namespace std; int main() { int n,i; cout << "n="; cin >> n; for (i = 1; i <= n; i++) cout << i << " "; return 0; }</pre>
<i>Exemplul 2: afișarea numerelor cuprinse între a și b (a<b), care au cifra unităților egală cu 3</i>	
Date de intrare: a, b Date de iesire: numerele cuprinse între a și b care au ultima cifra 3 Citeste a, b Pentru (x=a; x<=b; x=x+1) Daca (x%10==3) scrie x sf. Daca sf. Pentru	<pre>#include <iostream> using namespace std; int main() { int a, b, x; cout << "a="; cin >> a; cout << "b="; cin >> b; for (x = a; x <= b; x++) if (x%10 == 3) cout << x << " "; return 0; }</pre>

Citirea și prelucrarea pe rând a N numere

Citeste N

Pentru ($i=1; i \leq n; i++$)

Citeste X

Prelucreaza X

Sfarsit pentru

Efect:

Etapa 1. Se inițializează contorul i cu pasul 1 (variabila care numără pașii)

Etapa 2. Se verifică *daca pasul curent i este mai mic egal decât pasul final n*

- Dacă DA, se citeste un numar X, se prelucrează numărul X citit (prelucrarea depinde de cerințele problemei) apoi *contorul i crește cu 1 (trece la pasul următor)* și revenim la Etapa 2
- Dacă NU este îndeplinită, se oprește execuția structurii *Pentru*

În C++ (citire de la tastatură)

```
#include <iostream>
using namespace std;

int n, i, x;
int main()
{
    cin >> n;
    for (i = 1 ; i <= n; i++ )
    {
        cin >> x;
        //prelucrare x
    }
    return 0;
}
```

În C++ (citire din fișierul text **date.in)**

```
#include <fstream>
using namespace std;
ifstream fin ("date.in");
ofstream fout ("date.out");
int n, i, x;
int main()
{
    fin >> n;
    for (i = 1 ; i <= n; i++ )
    {
        fin >> x;
        //prelucrare x
    }
    return 0;
}
```

Exemplul 3: Se citesc prețurile a n produse. Să se calculeze și să se afișeze suma lor.

Algoritm	Program C++
Date de intrare: n Date de iesire: numerele cuprinse între 1 și n Citeste n Pentru ($i=1; i \leq n; i=i+1$) Citeste p $S \leftarrow S + p$ sf. Pentru scrie S	<pre>#include <iostream> using namespace std; int n, i, p, s; int main() { cout << "n="; cin >> n; for (i = 1; i <= n; i++) { cout << "p="; cin >> p; s = s + p; } cout << "suma=" << s; return 0; }</pre>

Aplicații:

1. Scrieți un program C++ care afișează toate numerele mai mici decât n, dat care au cifra zecilor egală cifra unităților.

2. Scrieți un program C++ care calculează suma cifrelor unităților pentru numerele cuprinse între a și b date.



Temă: problemele xyz, comoara, cadouri, psp, rachete, multimi, daruri, bancomat, bete1, prize, alo, pin, concurs3 de pe <http://campion.edu.ro/arhiva>