

Laborator 2 - Structuri repetitive

2.1. Structura repetitivă cu număr cunoscut de pași - FOR

Sintaxa:

for contor \leftarrow valoarea inițială, valoarea finală [, pas] **do**

 secvența de instrucțiuni;

end for

Execuție:

Pas 1: Se inițializează ciclul - contor= valoarea inițială;

Pas 2: Se verifică dacă expresia de continuare a ciclului este adevărată - contor < valoarea finală;

Pas 3: Dacă este adevărată se execută instrucțiune;

Pas 4: Se incrementează contorul ciclului - contor = contor + pas;

Pas 5: se continuă cu Pas 2

Observație: dacă [,pas] nu este prezent, pasul de incrementare este egal cu 1.

Exemplul 2.1. Să se calculeze suma $S = 1 + 2 + 3 + \dots + n$, unde n este un număr natural a cărei valoare este citită de la tastatură.

Comentariu. Vom utiliza un contor, i , care va lua succesiv valorile termenilor sumei, i.e., 1; 2; 3; ... ; n . La fiecare pas, vom adăuga valoarea contorului la suma s .

Pseudocod	Cod C
<pre> integer n, s, i; read n; s \leftarrow 0; for i \leftarrow 1, n do s \leftarrow s + i; end for write s; </pre>	<pre> int n,s,i; scanf ("%d", &n); s = 0; for(i=1; i<=n;i++) { s = s + i; } printf (" %d.",s); </pre>

2.2. Structura repetitivă cu număr necunoscut de pași și test inițial - WHILE

Sintaxa :

while expresie **do**

 secvența de instrucțiuni

end while

Execuție:

Această instrucțiune se poate traduce prin sintagma: cât timp o condiție dată este adevărată, se repetă un set de instrucțiuni. La fiecare pas (iterație) a buclei, dacă expresia este adevărată se execută o dată secvența de instrucțiuni.

Exemplul 2.2. Vom rezolva aceeași problemă de la exemplul 2.1.

Pseudocod	Cod C
<pre> integer n, s, i; read n; s \leftarrow 0; i \leftarrow 1; while i \leq n do s \leftarrow s + i; i \leftarrow i + 1; end while write s; </pre>	<pre> int n,s,i; scanf ("%d", &n); s = 0; i = 1; while(i <= n) { s = s + i; i = i + 1; // i++; } printf (" %d.",s); </pre>

Exemplul 2.3. Să se afișeze suma cifrelor unui număr întreg n .

Comentariu. Pentru a calcula suma cifrelor unui număr întreg, va trebui să obținem individual fiecare cifră. Deoarece nu știm câte cifre are numărul introdus de utilizator, vom extrage pe rând câte o cifră din număr și o vom adăuga la o sumă s , inițializată cu 0. Vom folosi formula $n\%10$, pentru a extrage ultima cifră a numărului n . După adăugarea ultimei cifre la sumă, aceasta poate fi eliminată din număr prin utilizarea formulei $n/10$, care, pentru n întreg, reprezintă numărul n fără ultima cifră.

Pseudocod	Cod C
<pre> integer n, s; read n; s ← 0; while n > 0 do s ← s + n%10; n ← n / 10; end while write s; </pre>	<pre> int n,s; scanf ("%d", &n); s = 0; while(n > 0){ s = s + n%10; n = n/10; } printf ("%d.",s); </pre>

Exemplul 2.4. Să se introducă de la tastatură numere întregi până la întâlnirea cifrei 0.

Pseudocod	Cod C
<pre> integer a; read a; while a != 0 do read a; end while </pre>	<pre> int a; scanf ("%d", &a); while(a != 0) { scanf ("%d", &a); } </pre>

2.3. Structura repetitivă cu număr necunoscut de pași și test final - DO ... WHILE

Sintaxa:

do

secvența de instrucțiuni

while *expresie*;

Vom rescrie Exemplul 3.4, utilizând instrucțiunea *do ... while*.

Pseudocod	Cod C
<pre> integer a; do read a; while a != 0 </pre>	<pre> int a; do{ scanf ("%d", &a); } while(a != 0); </pre>

Exemplul 2.5. Să se calculeze numărul de cifre ale unui număr întreg n .

Pseudocod	Cod C
<pre> integer n, nr_c; read n; nr_c ← 0; while n > 0 do nr_c ← nr_c + 1; n ← n / 10; end while write nr_c; </pre>	<pre> int n,nr_c; scanf ("%d", &n); nr_c=0; while(n > 0){ nr_c = nr_c + 1; n = n/10; } printf (" %d.",nr_c); </pre>

Exemplul 2.6. Să se calculeze răsturnatul(oglinditul, inversul) unui număr întreg n .

Pseudocod	Cod C
<pre> integer n, inv; read n; inv ← 0; while n > 0 do inv ← inv*10 + n%c; n ← n / 10; end while write inv; </pre>	<pre> int n,inv; scanf ("%d", &n); inv=0; while(n > 0){ inv = inv *10 + n%10; n = n/10; } printf (" %d.",inv); </pre>

Exemplul 2.7. Să se introducă de la tastatură numere întregi până la întâlnirea cifrei 0. Calculați valoarea maximă dintre numerele introduse.

Pseudocod	Cod C
<pre> integer a, max; read a; max ← a; while a != 0 do read a; if max < a then max ← a; end if end while write max;</pre>	<pre> int a,max; scanf ("%d", &a); max=0; while(a != 0) { scanf ("%d", &a); if (max < a) max = a; } printf ("%d",max);</pre>

Exemplul 2.8. Să se calculeze cel mai mare divizor comun a două numere întregi a și b.

Pseudocod	Cod C
<pre> integer a, b, r; read a, b; do r ← a % b; a ← b; b ← r; while r != 0 write "Gnmmdc=", a;</pre>	<pre> int a,b, r; scanf ("%d %d", &a, &b); do{ r = a % b; a = b; b = r; } while (r!=0); printf ("Gnmmdc= %d.",a);</pre>

Să se rezolve problemele de mai jos utilizând limbajul pseudocod.

Exercițiul 2.1. Să se afișeze mesajul formatat de mai jos, pentru n introdus de la tastatura. Spre exemplu dacă n=4, se afișează:

```

*
**
***
****
```

Exercițiul 2.2. Fie a și b două valori naturale, a > b. Să se simuleze împărțirea cu rest a lui a la b (prin scăderi repetate), a și b diferite de zero. Să se determine câtul și restul împărțirii.

Exercițiul 2.3. Să se calculeze suma primelor n numere naturale utilizând instrucțiunile "while", prima variantă și apoi "do while".

Exercițiul 2.4. Să se calculeze expresia $S=1+1*2+1*2*3+ \dots +1*2* \dots *n$, pentru n citit de la tastatură.

Exercițiul 2.5. Citiți de la tastatură un număr întreg, format cel puțin din 3 cifre. Afișați cifrele din care se compune numărul. Discuții: În ce ordine putem afișa cifrele?

Exercițiul 2.6. Determinați și afișați cea mai mare cifră care apare într-un număr natural, citit de la tastatură.

[Get the mobile app](#)