



Limbajul pseudocod și schema logică

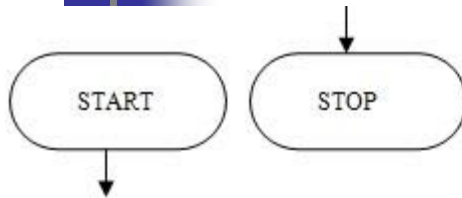
Pseudocodul este o forma de reprezentare a algoritmilor și ca orice limbaj folosește un grup de cuvinte cheie (*citește*, *scrie*, *dacă*, *atunci*, *cât timp*, *pentru*, etc.), organizate în construcții sintactice care traduc structurile fundamentale^[1], dar neavând o sintaxă rigidă ca un limbaj de programare. Limbajul pseudocod lasă la libera alegere a utilizatorului introducerea unor notații care să permită o mai bună exprimare.

Schema logică este un mijloc de descriere a algoritmilor prin reprezentare grafică. Fiecărui tip de operație îi este consacrată o figură geometrică (un bloc tip) în interiorul căreia se va înscrie operația din pasul respectiv, iar succesiunea operațiilor este indicată prin săgeți.

Blocuri utilizate în realizarea schemelor logice:

Lecția 3

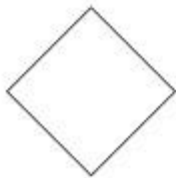
molnar1956@yahoo.com



Marchează începutul / sfârșitul unui algoritm reprezentat în schemă logică



Se utilizează pentru implementarea operațiilor de intrare / ieșire



Se utilizează pentru scrierea condițiilor în cazul structurilor alternativă sau repetitivă



Se utilizează pentru scrierea operațiilor de atribuire (atunci când o variabilă primește o valoare)



Se utilizează pentru conectarea liniilor care leagă diferitele figuri geometrice și marchează un punct de intersecție.



Legăturile dintre blocurile schemei logice se realizează numai prin intermediul săgeților, pentru a stabili sensul de parcurgere al algoritmului.



Pentru întocmirea unei scheme logice corecte sunt necesare respectarea unor anumite reguli după cum urmează:

- Orice schemă logică se întocmește de sus în jos;
- Legătura între blocurile schemei se face numai prin săgeți, pentru a stabili sensul de parcurgere;
- În cazul întâlnirii unui bloc de decizie se precizează deasupra liniilor de legătură când expresia

Vom prezenta în continuare structurile fundamentale, cu ajutorul cărora se poate scrie orice algoritm, în paralel schemă logică –limbaj pseudocod.

Structura liniară: poate fi

- *o operație de intrare / ieșire* (citire date de intrare /afișare rezultate)
- *o operație de atribuire*
- *un apel de funcție procedurală*
- *o combinație liniară* a acestora care se va executa necondiționat.



Operatia de citire:

Schema logică	Limbaj pseudocod
	citeste variabila1, variabila2,....variabila_n;

Operatia de afisare:

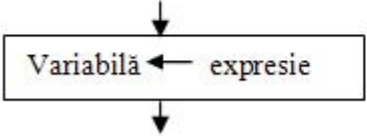
Schema logică	Limbaj pseudocod
	scrie variabila //afiseaza valoarea variabilei scrie " sir de caractere " //afiseaza mesajul din acolade scrie expresie aritmetica //afiseaza valoarea expresiei scrie orice combinatie de cele 3 variante

Lecția 3

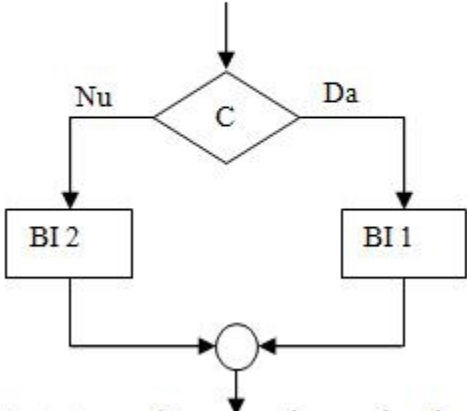
molnar1956@yahoo.com



Operatia de atribuire:

Schema logică	Limbaj pseudocod
	<pre>variabila <- expresie // valoarea expresiei este atribuita variabilei</pre>

Structura alternativa:

Schema logică	Limbaj pseudocod
	<pre>daca (conditie) atunci bloc de instructiuni 1; altfel bloc de instructiuni 2; _*</pre>

**Structura repetitiva cu test initial:**

Schema logică	Limbaj pseudocod
<pre>graph TD; Entry(()) --> C{C}; C -- Da --> BI[B.I.]; BI --> Entry; C -- Nu --> Exit(());</pre>	cat timp (conditie) executa Bloc de instructiune _*

Structura repetitiva cu test final:

Schema logică	Limbaj pseudocod
<pre>graph TD; Entry(()) --> BI[B.I.]; BI --> C{C}; C -- Da --> Entry; C -- Nu --> Exit(());</pre>	executa Bloc de instructiuni cat timp (conditie); _*

**Structura repetitiva cu numar fix de pasi:**

Schema logică	Limbaj pseudocod
<pre>graph TD; Entry(()) --> Init[i:=expr.1]; Init --> Cond{i <= expr.2}; Cond -- Nu --> Exit(()); Cond -- Da --> Instr[Bloc instr.]; Instr --> Inc[i:=i+pas]; Inc --> Cond;</pre>	Varianta cu pas pozitiv: pentru contor=expr1 pana la expr.2 cu pasul +const Bloc de instructiuni; _* Varianta cu pas negativ: pentru contor=expr1 pana la expr.2 cu pasul -const Bloc de instructiuni; _*



Descrieți etapele de rezolvare și scrieți un algoritm care să calculeze aria și perimetrul unui patrat de latură L , citită de la tastatură.

Rezolvare:

Analiza problemei:

Datele de intrare(Ce se cunoaște?): L , număr natural citit de la tastatură

Datele de ieșire(Ce se cere?): A - aria patratului și P -perimetrul patratului

Algoritm în limbaj natural:

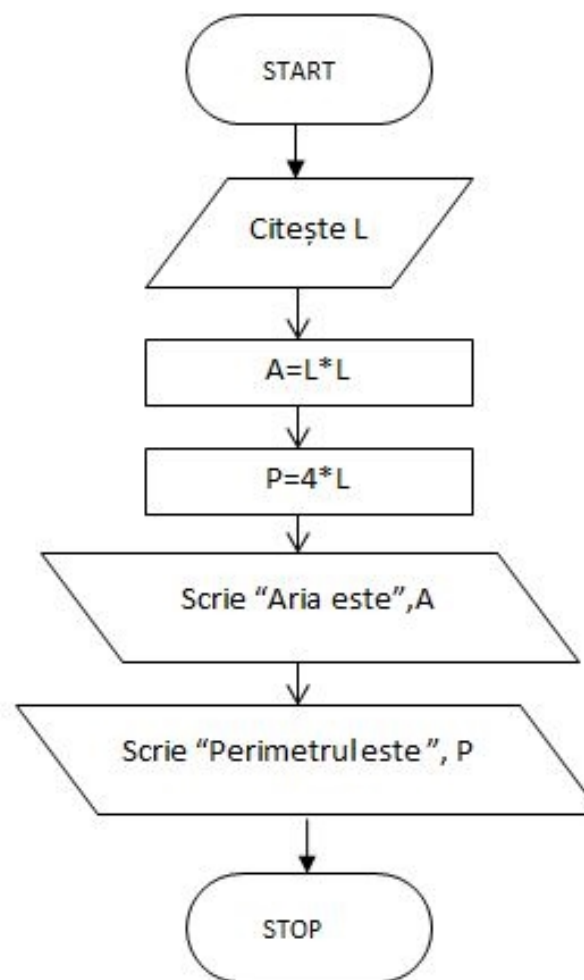
- citește de la tastatură valoarea variabilei L
- calculează cu ajutorul formulelor matematice aria și perimetru patratului
- afișează pe ecran valoarea ariei și a perimetrului



Algoritm in limbaj pseudocod

citeste L, numar natural
 $A = L * L$
 $P = 4 * L$
scrie "Aria este A=", A
scrie "Perimetrul este P=", P

Algoritm in schema logica





Descrieți etapele de rezolvarea și scrieți un algoritm care să calculeze media aritmetică dintre trei numere naturale.

Rezolvare:

Analiza problemei:

Date de intrare(Ce se cunoaște?): a, b, c trei numere naturale citite de la tastatură

Date de ieșire(Ce se cere?): ma - media aritmetică a celor trei numere

Algoritm în limbaj natural:

- citește valori pentru cele trei variabile a,b,c
- calculează media aritmetică a celor trei numere cu formula matematică

$ma = (a+b+c)/3$

- afișează media aritmetică ma



Algoritm in limbaj pseudocod

citește a, b, c numere naturale
 $ma = (a+b+c)/3$
scrie "Media aritmetica este ma=", ma

Algoritm in schema logica

