

# MODALITATI DE REPREZENTARE A STRUCTURILOR SECVENTIALE SI ALTERNATIVE PRIN BLOCURI GRAFICE

## SCHEMELE LOGICE

Sunt desene care respectă niște reguli stricte, fiind formate dintr-un număr de blocuri, conectate între ele, cu un singur punct de pornire (START) și un unic punct final (STOP);

Blocurile din care e alcătuită o schemă logică pot fi încadrate în 4 categorii:

- Blocuri START / STOP;
- Blocuri pentru operații de INTRARE / IEȘIRE;
- Blocuri de atribuire;
- Blocuri de decizie.

### Blocurile START / STOP

Fiecare schemă logică trebuie să aibă exact un bloc START și unul STOP

Ambele au o formă ovală ca în figura alăturată

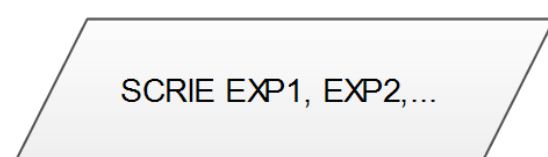
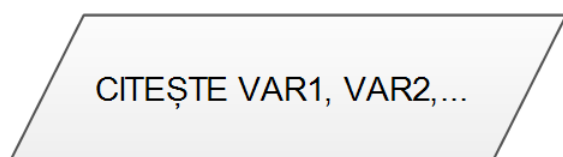


### Blocurile de INTRARE / IEȘIRE

În general algoritmi preiau date din exterior (date de intrare), pe care le prelucrează, furnizând în final rezultate (date de ieșire) ;

Preluarea datelor de intrare se face prin operații de “citire”, iar transmiterea rezultatelor (date de ieșire) prin operații de “afișare” ;

Pentru ambele tipuri de operații se folosesc blocuri în formă de paralelogram ca în figurile alăturate.



### Blocurile de atribuire

Correspond prelucrărilor efectuate în cadrul algoritmului;

Blocurile sunt dreptunghiuri ca în figura alăturată.



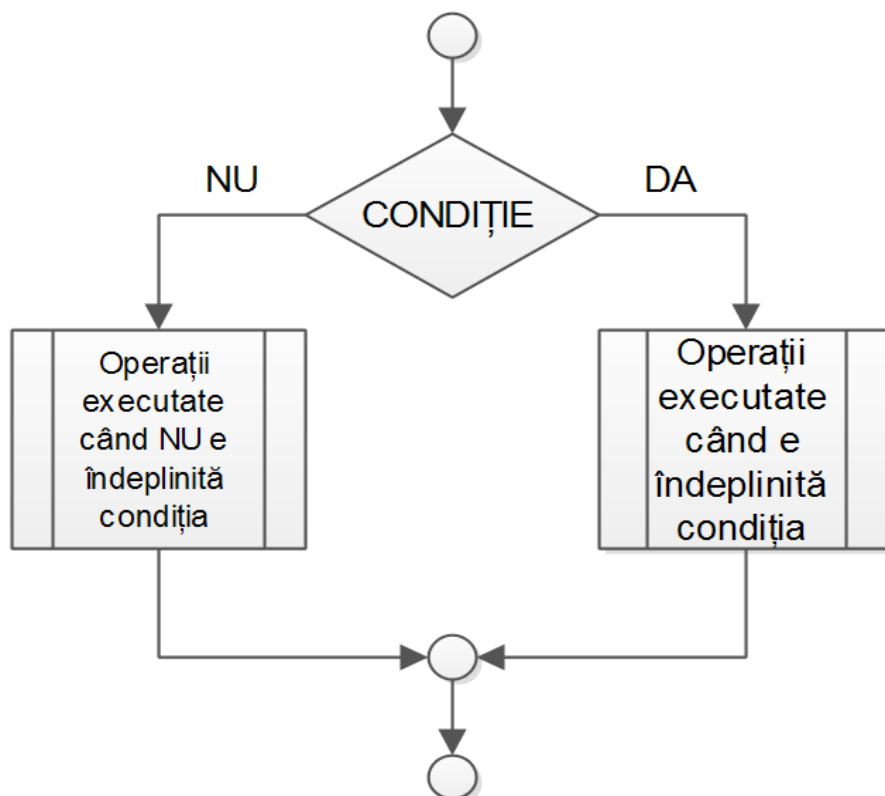
### Blocuri de decizie

Apar acolo unde algoritmul se ramifică în funcție de verificarea unei condiții (expresie logică, având valorile ADEVĂRAT sau FALS);

Întotdeauna ramura din dreapta va corespunde valorii “ADEVĂRAT” (“DA”, condiția este îndeplinită), iar cea din stânga valorii “FALS” (“NU”, condiția NU este îndeplinită);

Blocurile de decizie se folosesc și în cazul în care se dorește ca anumite operații să se repete cât timp o anumită condiție e îndeplinită; în astfel de situații, ramura “DA” se va întoarce în schemă undeva deasupra blocului de decizie. Apar acolo unde nu este necesară repetarea anumitor operații;

Sunt simbolizate prin romburi, în interiorul cărora apare câte o expresie logică (condiție).



## PSEUDOCODUL

- Este un limbaj de nivel înalt cu ajutorul căruia pot fi descriși algoritmi;
- Deși folosește convenții structurale asemănătoare cu cele ale unui limbaj de programare, pseudocodul a fost creat mai degrabă pentru a fi înțeles și interpretat de către oameni decât de către calculatoare;
- Un program pseudocod nu poate fi interpretat sau compilat de către un calculator, dar poate fi cu ușurință “tradus” în limbaj de programare de către un informatician.

Structuri de bază în pseudocod și echivalentul lor în scheme logice

- Pentru citirea datelor, în pseudocod se folosește instrucțiunea:

**citește variabile**

- Pentru afișarea expresiilor, se folosește:

**scrie expresii**

- Pentru atribuire, în pseudocod se folosește:

**variabilă ← expresie**

- Ca și în cazul schemelor logice, mai întâi se evaluează expresia din partea dreaptă a săgeții, apoi valoarea acesteia este reținută de variabila specificată în partea stângă.

- Echivalentul din pseudocod al blocurilor de decizie specifice schemelor logice este instrucțiunea “dacă”, având forma:

**dacă condiție**

    { instrucțiune 1 }

**altfel**

    { instrucțiune 2 }