

Algoritmi (recapitulare)

- datele cu care lucrează algoritmi : de intrare, de ieșire, de manevră, constante, variabile.

- datele au : *nume, tip, valoare*

numele

- conține litere mari și mici, cifre, _
- începe cu _ sau literă

tip – numerice, logice (adevărat sau fals), text, imagine, sunet

- **expresiile aritmetice** conțin date (operanzi) și operatori aritmetici + - * / %

/ împărțire cu rest (pentru numere întregi) sau fără rest (pentru nr raționale)

% restul împărțirii nr. întregi

Exemple : $13/4 - 5\%3$

$((2+4)*(5/2 +1) -3)/2$

- **atribuirea** este un pas al algoritmului prin care se încarcă o dată (variabilă) cu rezultatul unei expresii:

variabilă \leftarrow expresie ;

Exercițiu : a, b, c au valorile 5, 10, 12. Ce valori vor avea în urma secvenței

$a \leftarrow b+c$; $b \leftarrow c+a$; $c \leftarrow a+b$?

- **structura liniară (secvențială)** este un grup de pași dintr-un algoritm, care se execută o singură dată, în ordinea în care sunt scrise.

- structura liniară conține pașii : start, citește, scrie, atribuiri, stop

Etapele unui exercițiu algoritmic

1. se citește cu atenție problema
2. se identifică datele de intrare și de ieșire
3. se identifică formulele de calcul dacă este cazul
4. se descrie algoritmul în limbaj natural
5. se scrie algoritmul în limbajul specializat (pseudocod) sau cu blocurile unui program (Scratch, Blockly)
6. se execută algoritmul cu mai multe seturi de date de intrare

- Problemă rezolvată.

În curte sunt câteva pisici și cu 5 mai multe găini. Scrieți algoritmul care afișează câte picioare și câte capete sunt în curte.

Date de intrare : pisi,

Date de manevra : gaini

Date de ieșire : c, p

1. start
2. scrie "Introdu numarul de pisici :"
3. citeste pisi
4. $gaini \leftarrow pisi + 5$
5. $p \leftarrow gaini * 2 + pisi * 4$
6. $c \leftarrow gaini + pisi$
7. scrie "Numar de capete :", c
8. scrie "Numar de picioare :", p
9. stop