**ALGORITMI. NOTIUNI GENERALE**

**Noţiunea de algoritm. Proprietățile unui algoritm. Etapele obținerii unui program**

Dacă în matematică noţiunea cea mai importantă este numărul, **algoritmul** este conceptul fundamental al informaticii. Orice echipament de calcul poate fi considerat o maşină algoritmică.

**Algoritmul** reprezintă *o metodă de rezolvare a unei probleme (problemelor de un anumit tip) / o succesiune de pași bine definiți necesari pentru rezolvarea unei probleme.*

**Algoritmul** *este o succesiune finită de operaţii care se execută într-o ordine stabilită* astfel încât plecând *de la un set de date (numite* **datele problemei** *sau* **date de intrare** *=* **DI** *) ce îmdeplinesc anumite condiţii,* să obţinem *într-un interval de timp finit un set de valori (numite* **soluţiile problemei** *sau* **date de ieşire** *=* **DE** *) .*

Nu orice descriere pas cu pas a etapelor rezolvării unei probleme reprezintă un algoritm. Pentru a fi algoritm, o metodă de rezolvare a unei probleme trebuie să aibă **PROPRIETĂŢILE**:

* **Claritatea – proprietatea algoritmului prin care procesul de calcul este descris precis, fără ambiguități.**
* **Finitatea – proprietatea algoritmului de a se încheia într-un număr finit de paşi**
* **Generalitatea – proprietatea algoritmului de a rezolva o întreagă clasă de probleme, care sunt de acelaşi tip şi care diferă între ele numai prin datele de intrare.**
* **Realizabilitatea - paşii care compun algoritmul trebuie să reprezinte operaţii care se pot realiza cu resursele disponibile.**
* **Unicitatea** - pentru aceleaşi date de intrare se obţin totdeauna aceleaşi date de ieşire.
* **Eficienţa - este proprietatea unui algoritm de a se termina după un număr minim de paşi.**

Există totuşi algoritmi care sunt corecţi, clari, generali şi furnizează soluţia într-un timp finit, însă mai lung sau folosesc mai multă memorie decât alţi algoritmi. Aceasta înseamnă că atunci când elaborăm un algoritm, nu ne oprim la prima soluţie găsită. Vom încerca să gasim algoritmi care să dea soluţia într-un timp cât mai scurt, cu cât mai puţină memorie folosită. Cu alte cuvinte vom încerca să elaboram algoritmi **eficienţi**.

*Numim deci* ***eficienţă*** *- capacitatea algoritmului de a da o soluţie la o problema într-un timp de executie cât mai scurt (numărul de operații mai puține), folosind cât mai puţină memorie.*

Algorimul este ineficient şi dacă rezultatul se obţine într-un timp mai lung decât cel dorit sau permis.

Nerespectarea acestor caracteristici generale conduce la obținerea de algoritmi neperformanți, posibil infiniți sau nerealizabili.

Observația1. Nu orice problemă admite un algoritm de rezolvare.

Observația2. Doi algoritmi sunt echivalenți când pentru aceleași date de intrare se obțin aceleași date de ieșire.

Instrucțiune - unitate sintactică a unui program ce indică operația de efectuat.

**Etapele rezolvării problemelor/Etapele obținerii unui program**

1. **Analiza problemei, pentru a identifica datele de intrare şi de ieşire.**
2. **Elaborarea algoritmului de rezolvare a problemei (găsirea metodei de rezolvare a problemei)**
3. **Transpunerea algoritmului într-un limbaj de programare.**
4. **Testarea programului și corectarea erorilor (se va folosi o mulțime de seturi de date care să prevadă toate situațiile care pot să apară în exploatare a programului).**