



SUBPROGRAME C++

1. Introducere

Subprogramele sunt părți ale unui program, identificabile prin nume, care se pot activa la cerere prin intermediul acestor nume.

Prezența subprogramelor implică funcționarea în strânsă legătură a două noțiuni: **definiția** unui subprogram și **apelul** unui subprogram.

Definiția unui subprogram reprezintă de fapt descrierea unui proces de calcul cu ajutorul variabilelor virtuale (parametri formali) iar **apelul** unui subprogram nu este altceva decât execuția procesului de calcul pentru cazuri concrete (cu ajutorul parametrilor reali, (efectivi, actuali)).



2. Structura unui subprogram C++

Un subprogram (funcție) are o *definiție* și atâtea *apeluri* câte sunt necesare.

Definiția unei funcții are forma generală:

```
tip_returnat nume_funcție (lista parametrilor formali)
{
    instrucțiune; // corpul funcției
}
```



Tip_returnat	Reprezintă tipul rezultatului calculat și returnat de funcție și poate fi: int, char, char*, long, float, void, etc. În cazul în care tipul rezultatului este diferit de void, corpul funcției trebuie să conțină cel puțin o instrucțiune return. Înstrucțiunea return va specifica valoarea calculată și returnată de funcție care trebuie să fie de același tip ca și tip_returnat.
Nume_funcție	Reprezintă numele dat funcției de către cel ce o definește, pentru a o putea apela.
Lista_parametrilor_formali	Reprezintă o listă de declarații de variabile separate prin virgulă. Această listă poate să fie și vidă.
Instrucțiune	Este o instrucțiune vidă, simplă sau compusă.