**Transmiterea parametrilor**

**Parametri formali**  pot fi de două tipuri:

1. Parametri formali de **tip valoare**
2. Parametri formali de **tip referință/adresă**

**i) Parametri formali de tip valoare**

* Se declară: ***tip1 p1, tip2 p2, …, tipn pn***
* Reprezintă **date de intrare**  pentru subproblema rezolvată de funcție.
* Substituția/înlocuirea la apel a parametrilor formali de tip valoare cu parametri efectivi corespunzători se numește  **apel/transmitere prin valoare**.
* La transmiterea parametrilor prin valoare se lucrează pe o **copie** a parametrului efectiv corespunzător (copie făcută în stiva limbajului). De aceea, orice modificare adusă unui astfel de parametru **se pierde** la ieșirea din funcție. Deci, parametri efectivi corespunzători **ies nemodificați** din funcție.
* Parametri efectivi corespunzători parametrilor formali de tip valoare pot fi: **constante, variabile și expresii**.

**ii) Parametri formali de tip referință/adresă**

* Se declară: ***tip1 &p1, tip2 &p2, …, tipn &pn***
* Reprezintă **date de ieșire/rezultate**  pentru subproblema rezolvată de funcție.
* Substituția/înlocuirea la apel a parametrilor formali de tip referință/adresă cu parametri efectivi corespunzători se numește  **apel/transmitere prin referință/adresă**.
* La transmiterea parametrilor prin referință funcția lucrează direct la adresa de memorie a parametrului efectiv corespunzător. De aceea, orice modificare adusă unui astfel de parametru **rămâne vizibilă** la ieșirea din funcție. Deci, parametri efectivi corespunzători **ies modificați** din funcție.
* Parametri efectivi corespunzători parametrilor formali de tip adresa pot fi **doar nume de variabile.**

**Obs.** Parametri de tip tablou (vectori/matrice) sunt transmiși implicit prin referință (se transmite adresa de început a tabloului deoarece, dacă a este numele tabloului a ⬄ &a[0]).

**Ex.** Ce afișează urmatorul program?

int x, y

**; //** variabile globale

void F (int x, int &y)

{ x=x+2; y--;

cout<<x<<” “<<y<<’\n’; // 12 6

}

int main()

{ x=10; y=7;

F(x,y);

cout<<x<<” “<<y<<’\n’; // 10 6

return 0;

}