

În C, putem *declara variabile* de următoarele tipuri:

elementare: int, char, double, float, unsigned etc.

inclusiv *adrese* de orice tip: *tip* *var

tablouri: elemente de același tip

structuri: elemente cu tipuri posibil diferite

Un șir are tipul `char *` (e reprezentat prin adresa șirului)

În C (și în limbaje de programare în general) putem folosi valori:

în *atribuire* ... = v

ca *parametri* la funcții: ... f(v, ...) ...

operatorii din expresii sunt similari funcțiilor

ca *rezultat returnat* din funcție: return v;

În C, putem folosi astfel valori *elementare* (incl. *adrese*) și *structuri*

NU putem folosi astfel un *tablou* (ca tot unitar, valoare compusă)

Când folosim un tablou, folosim de fapt *adresa* lui

Când folosim (atribuim, transmitem, returnăm) valori (elementare sau structuri) se creează *copii* ale lor

transmiterea parametrilor se face *prin valoare*

o funcție NU poate modifica o variabilă a cărei valoare e transmisă ca parametru

Un *pointer* (*referință*) reprezintă unic un obiect, prin *adresa* lui mai multe structuri de date se pot referi la același obiect
modificarea obiectului e văzută peste tot