

Lecția 9

Operatori de atribuire

Operatorii de atribuire sunt operatori binari care permit modificarea valorii unei variabile. Limbajul C++ are un singur operator de atribuire simplu (=) și 10 operatori de atribuire compuși cu ajutorul operatorului de atribuire simplu.

Operația de atribuire simplă are următoarea sintaxă:

variabilă = expresie;

Efect:

- se calculează valoarea expresiei și se obține un rezultat;
- rezultatul se memorează la adresa variabilei.
- Efectul acestei operații este întotdeauna de la dreapta la stânga

Atenție!

- A nu se confunda operația de atribuire cu operația matematică desemnată prin același simbol "=". De exemplu daca avem declarația int a=10, operația a=a+1; în informatică este perfect corectă și are ca semnificație schimbarea valorii variabilei a la valoarea 11. Din punct de vedere matematic relația a=a+1 este o relație gresită.

Operația de atribuire compusă are următoarea sintaxă:

variabilă operator = expresie;

unde operatorul aparține mulțimii {*, /, %, +, -, <<, >>, &, |, ^}.

Expresia din partea dreaptă a semnului "=" poate fi la rândul ei tot o operație de atribuire, cu alte cuvinte se pot scrie operații de atribuire înlănțuite respectând următoarea sintaxă:

variabilă_1=variabila_2=...=variabila_n=expresie;

Exemplu:

Fie două variabile a=30 și b=2 de tip întreg int atunci:

Operatorul de incrementare și de decrementare

Operatorii de incrementare/ decrementare sunt operatori unari care au drept scop creșterea/micșorarea valorii unei variabile cu 1.

Operator

++

--

Denumire

incrementare

decrementare

Forma generală de utilizare a acestor operatori este:

Operatori condiționali

Operatorii condițională sunt "?" și ":" și se utilizează numai împreună. Sintaxa generală a unei expresii condiționale este:

expresie_1 ? expresie_2 : expresie_3

Efect:

- se evaluează expresie_1.
- Dacă expresie_1 are o valoare diferită de 0 atunci valoarea expresiei condiționale este egală cu valoarea expresiei_2.

- Dacă expresie_1 are o valoare egală cu 0 atunci valoarea expresiei condiționale este egală cu valoarea expresiei_3.

Exemplu: expresia $x < y ? x : y$ determină valoarea minimă dintre x și y.

Operatorul de adresă(referință)

Este un operator unar care permite determinarea adresei zonei de memorie în care este stocată o variabilă. Acest operator este & și se utilizează după următoarea sintaxă generală:

&variabilă

Operatorul de conversie explicită

Operatorul de conversie implicită este un operator unar care permite conversia fortată a tipului unei expresii la un tip specificat. Forma generală este:

(tip nou)expresie

Exemplu:

1. Dacă considerăm două variabile de tip întreg int a=20 și b=15 și dorim să calculăm media aritmetică dintre cele două variabile scriem:

(float) (a+b)/2 rezultatul expresiei fiind 17.5

În lipsa conversiei rezultatului operatorul / va calcula câtul împărțirii sumei (a+b) la doi iar rezultatul obținut va avea valoarea 14.

1. Dacă considerăm variabila de tip int a=70 și dorim să calculăm valoarea expresiei a*1000 rezultatul obținut va depăși limita superioară admisă tipului int, adică 32767 și rezultatul afișat va erona și va avea valoarea 4464. Pentru a obține rezultatul corect atunci rezultatul expresiei date trebuie fortate la tipul long și astfel expresia dată își va modifica forma astfel : (long)a*1000

Operatorul de determinare a dimensiunii unei variabile sau expresii

Operatorul de determinare a dimensiunii în octeți a unei variabile sau a unei expresii necesare memorării acesteia este sizeof. Forma generală este:

sizeof(tip) sau sizeof(expresie)

Exemplu: Daca avem o variabilă de tip int a=70, o variabilă b de tip float și o variabila c de tip char atunci:

Evaluarea expresiilor

Evaluarea unei expresii presupune determinarea valorii acelei expresii, prin înlocuirea în expresie a fiecărei variabile cu valoarea ei și a fiecărei funcții cu valoarea returnată de funcția respectivă și efectuarea operațiilor precizate de operatori. În evaluarea unei expresii se ține cont de :

- Existența parantezelor
- Asociativitate
- Prioritatea operatorilor

Etape în evaluarea unei expresii:

- Se vor calcula în primul rând expresiile din interiorul parantezelor, începând cu cele mai interioare
- În situația în care avem o expresie fără paranteze atunci ordinea de evaluare este dată de prioritatea operatorilor folosiți
- În cazul în care avem mai mulți operatori de aceeași prioritate , se va ține cont de asociativitatea operatorilor.

În limbajul C++ operatorii se vor asocia de la stânga la dreapta, excepție făcând operatorul de atribuire, operatorii unari și condiționali care se asociază de la dreapta la stânga. În situația în care operanții nu au același tip, atunci pe parcursul evaluării expresiei se vor realiza în mod automat o serie de conversii隐式的 de tip după urmatoarea regulă: „*operatorul care are un domeniu de valori mai mic va trece în urma conversiei implicite la tipul operandului care are cel mai mare domeniu de valori*”.

Tabelul cu prioritatea tuturor operatorilor

