

# EXPRESII

## 1. Conținutul lucrării

Scopul lucrării este prezentarea noțiunii de expresie în limbajul C/C++, a operatorilor și a modului de evaluare a unei expresii.

### 1. Considerații teoretice

#### 1.1 Definirea expresiei

O expresie este formată dintr-un operand sau mai mulți operanzi legați prin operatori.

Un operand poate fi:

- o constantă;
- o constantă simbolică;
- numele unei variabile simple;
- numele unui tablou;
- numele unei structuri;
- numele unui tip;
- o variabilă indexată;
- numele unei funcții;
- referința la elementul unei structuri;
- apelul unei funcții;
- expresie inclusă în paranteze rotunde.

Unui operand îi corespunde o valoare și un tip.

Operatorii pot fi unari sau binari.

La evaluarea unei expresii trebuie să se țină seama de:

- prioritățile operatorilor;
- asociativitatea operatorilor de aceeași prioritate;
- regula conversiilor implicite.

#### 1.2 Operatori

Operatorii limbajului C/C++ sunt împărțiți în următoarele clase:

- operatori aritmetici:
  - operatori unari: +, -
  - operatori binari multiplicativi: \*, /, %
  - operatori binari aditivivi: +, -
- operatori relaționali: <, <=, >, >=
- operatori de egalitate: =, !=

- operatori logici: !, &&, ||
- operatori logici pe biți: ~, <<, >>, &, ^, |
- operatori de atribuire: =, /=, \*=, %=, +=, -=, <<=, >>=, &=, ^=, |=
- operatori de incrementare: ++, --
- operatori de forțare tip: (tip) operand
- operatori dimensiune: sizeof
- operatori adresă: &
- operatori paranteză: (), []
- operatori condiționali: ?, :
- operatorul virgulă: ,
- operatorul de dereferențiere: \*
- operatorul de acces la componenta unei structuri: ., ->

Observație: În limbajul C++ s-au mai introdus câțiva operatori:

- operatorul de rezoluție: ::
- operatorul pentru tipul referință: &
- operatorul de alocare/dealocare dimensiune a memoriei heap: new/delete

Prioritățile operatorilor, în ordine descrescătoare, sunt prezentate în tabelul de mai jos:

Prioritățile	Operatori
1	() [] • ->
2	+(uniar) -(uniar) &(uniar) *(uniar) ++ -- (tip) sizeof !~
3	*(binar) / %
4	+(binar) -(binar)
5	<< >>
6	< <= > >=
7	= = !=
8	&(binar)
9	^
10	
11	&&
12	
13	? :
14	= <<= >>= += -= *= /= %= &= ^=  =
15	,

Operatorii din aceeași linie au aceeași prioritate.

În ceea ce privește asociativitatea operatorilor, se precizează că ei se asociază de la stânga la dreapta cu excepția operatorilor unari, condiționali și de atribuire, care se asociază de la dreapta la stânga.

### 2.3. Regula conversiilor implicite

Regula conversiilor implicite constă în următoarele:

- dacă un operator binar se aplică la operanzi de același tip, atunci rezultatul va avea tipul comun al operanzilor;
- dacă un operator binar se aplică la doi operanzi de tipuri diferite, atunci operandul de tip inferior se convertește implicit spre tipul superior al celuilalt operand, iar rezultatul va avea tipul superior.

Ordinea descrescătoare a priorității tipurilor este următoarea:

- long double;
- double;
- float;
- unsigned long;
- long;
- int.

### 3. Mersul lucrării

- 3.1. Să se scrie un program pentru calculul valorii  $z = x**y$ ,  $x$  și  $y$  fiind de tipul double.
- 3.2. Explicați deosebirea între împărțirea reală și cea întreagă.
- 3.3. Să se scrie un program care citește un unghi în grade sexagesimale și calculează valoarea funcției sinus, cosinus și tangentă.
- 3.4. Să se scrie un program care citește un număr natural în intervalul [1600, 4900] ce reprezintă un an și verifică dacă este bisect sau nu.
- 3.5. Folosind expresii condiționale, să se scrie un program care citește valoarea reală a lui  $x$  și calculează valoarea funcției:

$$f(x) = \begin{cases} x^2 + 4x + 4 & \text{daca } x < -2 \\ 0 & \text{daca } x = 0 \\ x^2 + 5x & \text{daca } x > -2 \end{cases}$$

- 3.6. Să se scrie un program care citește un număr real  $x$ , ce reprezintă măsura unui unghi în radiani, și îl transformă în grade, minute și secunde sexagesimale.
- 3.7. Să se scrie un program care simulează funcționarea unui numărător de tip ceas (se indică ora, minutul și secunda).
- 3.8. Scrieți un program pentru a indica numărul de octeți ocupați în memorie de tipurile de date din limbajul C/C++.

3.9. Converteți în binar, prin calcul, anul nașterii și anul curent.

Arătați cum se reprezintă ca o dată de tip `int`. Efectuați operațiile de deplasare stânga cu 4 biți, dreapta cu 2 biți, complement față de 1 asupra lor, precum și operațiile pe biți `&`, `^`, `|`, având ca operanzi cele două date. Scrieți un program pentru a verifica corectitudinea calculelor dumneavoastră.

3.10. Scrieți un program ce efectuează operații aritmetice asupra a două date de tip întreg și real. Executați-l pentru valori care conduc la rezultat în afara limitelor de reprezentare internă. Ce se întâmplă în acest caz?