



LIMBAJE DE PROGRAMARE

INSTRUCȚIUNI / COMENZI PENTRU IMPLEMENTAREA ÎN LIMBAJ DE PROGRAMARE
A STRUCTURII ALTERNATIVE

INSTRUCȚIUNI/COMENZI PENTRU IMPLEMENTAREA ÎN LIMBAJ DE PROGRAMARE A STRUCTURII ALTERNATIVE.

Instrucțiunea IF este cea mai utilizată structură alternativă. ea are două forme:

```
IF (expresie)  
    Instrucțiune 1
```

Mod de execuție

Pas 1: Se evaluează expresie,

Pas 2: **dacă** valoarea ei este diferită de zero se execută **instrucțiune1**

Pas 3: se continuă cu instrucțiunea următoare instrucțiunii if

```
IF (expresie)  
    Instrucțiune 1  
ELSE  
    Instrucțiune 2
```

Pas 1: Se evaluează expresie,

Pas 2: **dacă** valoarea ei este diferită de zero se execută **instrucțiune1** **altfel** se execută **instrucțiune2**

Pas 3: se continuă cu instrucțiunea următoare instrucțiunii **if**

În cadrul expresiilor logice, pot să apară **operatori relaționali** și **operatori logici**.

OPERATORI RELAȚIONALI

operator	descriere
==	egalitate
!=	diferit
<	mai mic
<=	mai mic sau egal
>	mai mare
>=	Mai mare sau egal

OPERATORI LOGICI

		&& - și logic	- sau logic
a	b	a && b	a b
1	1	1	1
1	0	0	1
0	1	0	1
0	0	0	0

Negatie logică	
a	!a
1	0
0	1

- **Exemplu** pentru variabilele $a = 5$, $b = 7$

expresia $(a > 4) \ \&\& \ (b < 8)$ este adevărată (1) - atât prima cât și a doua expresie sunt adevărate;

expresia $(a > 4) \ || \ (b > 8)$ este adevărată (1) – deoarece prima expresie e adevărată.

- Evaluează valoarea de adevăr a expresiilor din tabelul de mai jos, pentru variabilele $a = 7$, $b = 9$

$(a < 8) \ \ (b < 10)$	$a + 5 > b$	$(a < b) \ \ (b > a)$
$(a - b > 0) \ \&\& \ (b > 3)$	$(a / 10 == b / 10)$	$(a > 3) \ \&\& \ (b > 6)$

INSTRUCȚIUNEA DE SELECȚIE MULTIPLĂ SWITCH

FORMA GENERALĂ

```
switch (expresie_intreaga)
{
    case val1:
        /* grupul de instructiuni 1 */
        break;
    case val2:
        /* grupul de instructiuni 2 */
        break;
        /* ... */
    case valn:
        /* grupul de instructiuni n */
        break;
    default:
        /* grupul de instructiuni n+1 */
}
```

FUNCȚIONARE

Semnificația instrucțiunii este: Se evaluează expresia **expresie_intreaga** care trebuie să aibă o valoare întreagă.

- Avem următoarele situații:
Dacă valoarea expresiei este egală cu **val₁**, atunci se execută grupul de instrucțiuni 1.
- Dacă valoarea expresiei este egală cu **val₂**, atunci se execută grupul de instrucțiuni 2.
.....
- Dacă valoarea expresiei este egală cu **val_n**, atunci se execută grupul de instrucțiuni n.
- Dacă valoarea obținută în urma evaluării expresiei nu este egală cu nici una dintre valorile **val₁, ..., val_n**, atunci se execută grupul de instrucțiuni n+1.

OBSERVAȚII

1. La instrucțiunea **switch** grupurile de instrucțiuni de pe ramuri nu trebuie să fie delimitate cu acolade. Nu este greșit însă dacă le delimităm totuși cu acolade.
2. După fiecare grup de instrucțiuni punem în general instrucțiunea **break**. În lipsa instrucțiunii **break** se execută și instrucțiunile de pe ramurile de mai jos până la sfârșitul instrucțiunii **switch** (inclusiv cele de pe ramura **default**) sau până se întâlnește primul **break**. Instrucțiunea **break** întrerupe execuția instrucțiunii **switch** și a celor repetitive (**for**, **while** și **do ... while**).
3. Ramura **default** poate lipsi.
4. Dacă există ramura **default**, nu este obligatoriu să fie ultima.
5. Valorile **val₁**, **val₂**, ..., **val_n** trebuie să fie **constante întregi** și distincte două câte două.