

Curs 2 - continuare

Fiecare caracter char are două forme: forma de char '@' forma numerică ASCII

cod ASCII

char [-128, 127]

'a' → 97

⋮

'z' → 122

'\0' char 0

char caracter;

printf("..."),

scanf("%c", &caracter);

'\0' caracterul zero, nu e valoare întregă 0, ci este întreg 48

Există o echivalență între fiecare caracter și codul lui ASCII.

space / spatie → 32

Enter → \n → \r \lf → 13, 10

↙ carriage return
↘ line feed

cr
lf
a b c d e f g h

char caract;

printf("Dati un caracter:");

scanf("%c", &caract);

printf("%c %d", caract, caract);

sub formă de int
sub formă de caracter

Ce se întâmplă dacă dorim să citim amestecat atât tipuri de date numerice, cât și caractere?

int a, b, c;

char optiune;

printf("Dati 3 valori intregi si un char");

scanf("%d%d%d%c", &a, &b, &c, &optiune);

Program 123 <Enter> 456 <Enter> 789 <Enter>
bufferul de stocare a memoriei de stocare

Obs: scanf / cum neglijează caracterele albe / intrate space (tab, space, enter). doar dacă urmează după citire numerică, în cazul citirii caracterelor, dacă urmează enter, tab, space el se stochează în variabile caracter

123 <space> 456 <space> 789 <Enter> C

scanf("%d%d%d", &a, &b, &c);

getchar(); ← SE ÎNCETEȘTE ENTER-UL

scanf("%c", &optiune);

fflush(stdin); // golește bufferul de stocare

~~123 <sp> <tab> <Enter> / 456 <sp> <sp> <tab> <tab> 789 <Enter>~~
← golire buffer
fflush(stdin)

Instrucțiuni de ramificare

if (condiție logică) // NU SE PUNE ;

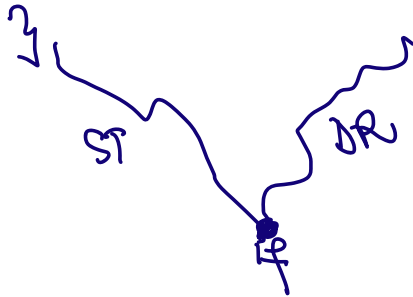
condiție
expresie / variabilă

if (a + b * 3 - 10)

expresie 0 → false
expresie != 0 → true

if (condiție)
instrucțiune1;

if (condiție)
{
instrucțiune1;
instrucțiune2;
...
}



if (condiție)

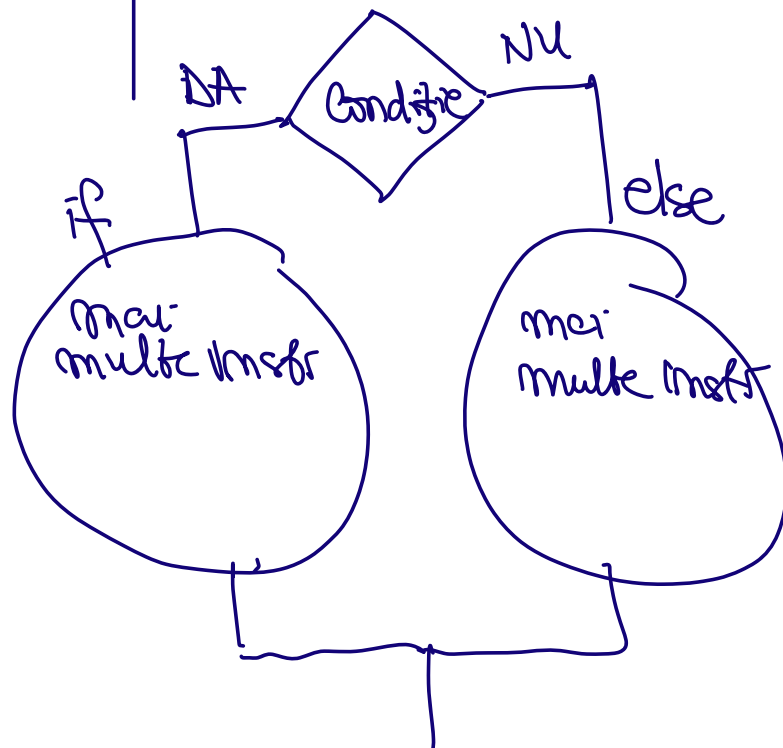
{
...
}

{

else // condiție falsă

{
...
}

{



Operatorul ternar: 3 operanzi / 2 operații

? : condiție ? instr DA : instr NU ;

```

int a=10, b=20, m;
if(a>b) m=a;
else m=b;

```

$m = (a > b ? a : b);$

Macro cu define și operator

```

#define MAX(mr1, mr2) (mr1 > mr2 ? mr1 : mr2)

```

```

int main()
{
    int a=1, b=2;
    int m = MAX(a, b);
}

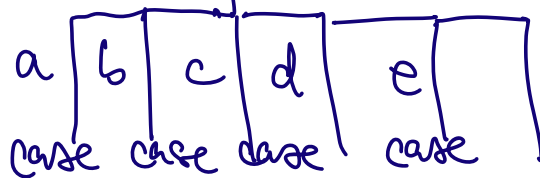
```

$1 > 2 ? 1 : 2$

Instrucțiune condițională cu mai multe ramuri

switch

switch



default, dacă nu e niciunul dintre celelalte

