

# Programarea calculatoarelor

## Limbajul C



### CURS 3



### Instrucțiuni C



# Cuprins

---

- Instrucțiuni
  - Recapitulare curs 2
  - while
  - do.. while
  - for
  - switch
  - break
  - continue
  - goto
  - return

# Instrucțiuni C

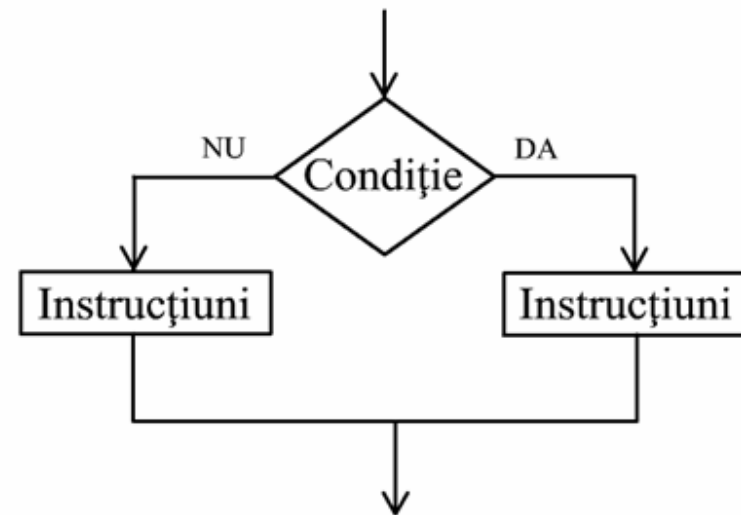
---

- Instrucțiunea vidă ;
- Instrucțiunea de atribuire `a = 3`
- Instrucțiunea compusă (bloc)

{ instr1; instr2; etc }

- Instrucțiunea if

```
if (expresie) Instrucțiune1;  
else Instrucțiune2
```



# Probleme propuse

---

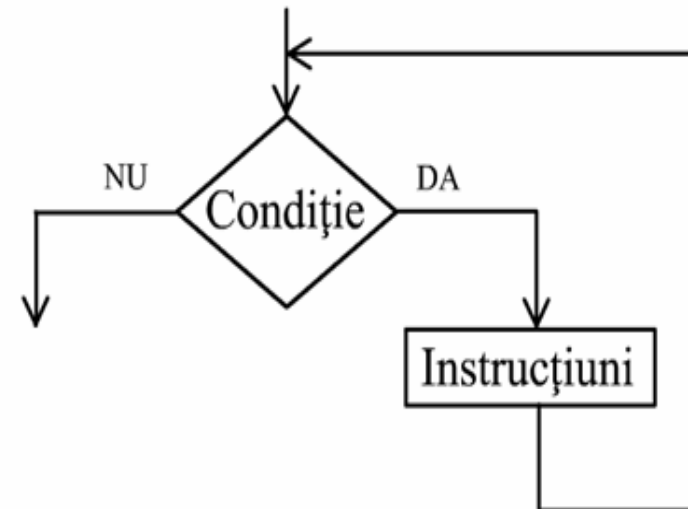
1. Interschimbul valorilor a două variabile a și b.
2. Rezolvarea ecuației de grad 2:  $ax^2+bx+c=0$ .
3. Se citesc de la tastatură 3 numere întregi reprezentând lungimile laturilor unui triunghi. Să se calculeze și să se afișeze aria triunghiului.
4. Să se afișeze în ordine crescătoare valorile a 3 variabile a, b și c.
5. Program care verifică dacă un caracter dat este o literă (fie mare, fie mică)

# Instrucțiunea while

- while ( expresie )  
instrucțiune;

- Exemplu:

```
#include<stdio.h>
void main(void)
{
    int a=18, b=12, r;
    while ( a%b>0 ) {
        r=a%b;
        a=b;
        b=r;
    }
    printf("cmmdc este %d\n",b);
}
```



# Instrucțiunea while

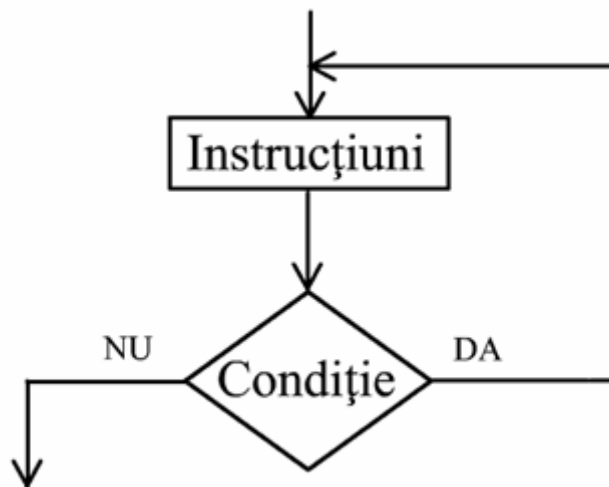
---

- Exemplu:

```
#include<stdio.h>
void main(void)
{
    int a=18, b=12, r;
    while (r=a%b){
        a=b;
        b=r;
    }
    printf("cmmdc este  %d\n",b);
}
```

# Instrucțiunea do while

```
do  
    instrucțiune;  
while(expresie);
```



Exemplu:

```
#include<stdio.h>  
void main(void)  
{  
    int s=0,i=0,n;  
    printf("introduceti n:\n");  
    scanf("%d",&n);  
    do{  
        s+=i;  
        i++;  
    }while (i<=n);  
    printf("suma primelor %d nr. nat. este  
           %d\n",n,s);  
}
```

# Instrucțiunea do while

---

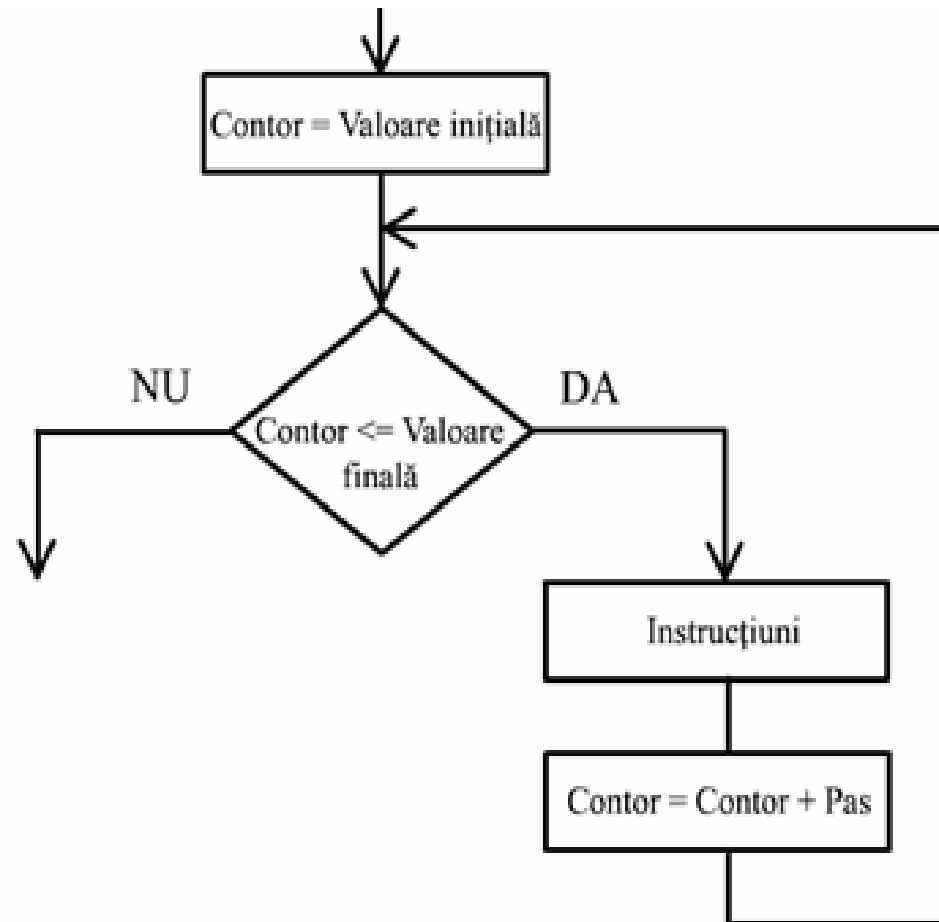
- Exemplu - citire repetată până când n ia o valoare între 1 și 1000

```
do {  
    printf("n= ");  
    scanf("%d",&n);  
} while (n>1000 || n<=0)
```



# Instrucțiunea for

for (conditie\_start; conditie\_continuare; re-evaluare)  
instrucțiune;



# Instrucțiunea for

---

- Exemplu:

```
#include<stdio.h>
void main(void)
{
    int n, fact=1, i;
    printf("introduceti n: ");
    scanf("%d", &n);
    for(i=1; i<=n; i++) fact*=i;
    printf("%d! = %d\n",n,fact);
}
```

# Instrucțiunea for

---

- Oricare din expresii poate fi vidă
- Dacă lipsește condiția de continuare – ciclu infinit
- Nu pot lipsi ;
- Exemplu:
  - for ( i=10; ;i--) instructiune
  - for ( ; ; ) instructiune
  - Echivalent cu: while(1) instructiune

# Instrucțiunea for

---

- Expresii compuse – folosind operatorul ,
- Exemplu

```
#include<stdio.h>
void main(void)
{
    int n, fact, i;
    printf("introduceti n: ");
    scanf("%d", &n);
    for ( fact=1, i=1; i<=n; i++) fact*=i;
    //sau :
    // for ( fact=1, i=1; i<=n; fact*=i, i++);
    printf("%d! = %d\n",n,fact);
}
```

# Instrucțiunea *switch*

---

```
switch (expresie) {  
    case c1: instr1;  
    [case c2: instr2;]  
    ...  
    [default: instr_default;]  
}
```

- c1, c2, etc – constante sau expresii întregi (inclusiv char)

# Instrucțiunea *switch*

---

- Exemplu: determinare semn număr prin citirea primului caracter c (care este semnul - sau cifra)

```
switch (c=getchar()) {  
    case '-': semn =1; c=getchar(); break;  
    case '+': c=getchar();    // si semn=0  
    default: semn=0;        // semn implicit  
}
```

# Instrucțiunea *switch*

---

```
switch (expresie) {  
    case c1: instr1; break;  
    [case c2: instr2; break; ]  
    ...  
    [default: instr_default;]  
}
```

# Instrucțiunea *switch*

---

```
char c; // c poate fi +,-,*,/  
switch ( c=getchar() ) {  
    case ' + ': r =a + b; break;  
    case ' - ': r =a - b; break;  
    case ' * ': r =a * b; break;  
    case ' / ': r =a / b; break;  
    default: printf("Eroare!\n");  
}
```



# Alte instrucțiuni C

---

- Instrucțiunea break → ieșire forțată din ciclu sau switch
- Salt după instrucțiunea din care a ieșit
- Instrucțiunea continue → continuă cu următoarea iterație a ciclului = salt la prima instrucțiune din ciclu)
- Instrucțiunea goto → saltul la o instrucțiune precedată de o etichetă

```
goto eticheta;  
eticheta: instrucțiune;
```

# Exemplu break

---

```
// verifică dacă un număr dat n este prim
for (k=2; k<n;k++)
    if ( n%k==0) break;
if (k==n) printf ("prim \n");
else printf ("neprim \n");
```

# Validare citire (ore, minute, secunde) – exemplu continue

---

```
int h,m,s;           // h=ore, m=min, s= sec
int corect=0;
while ( ! corect ) {
    printf (" ore, minute, secunde: ");
    if ( scanf("%i%i%i", &h, &m, &s) !=3 ) {
        printf (" eroare in datele citite \n");
        fflush(stdin); continue;           // salt peste instruct. urmatoare
    }
    if ( h <0 || h >24 ) {
        printf (" eroare la ore \n");
        fflush(stdin); continue;         // salt peste instruct. urmatoare
    }
    if ( m<0 || m > 59 ) {
        printf (" eroare la minute \n");
        fflush(stdin); continue;        // salt peste instruct. urmatoare
    }
    if ( s<0 || s > 59 ) {
        printf (" eroare la secunde \n");
        fflush(stdin); continue;       // salt peste instruct. urmatoare
    }
    corect=1;
}
```

# Instrucțiunea return

---

- return;
- return expresie;

- Exemplu:

```
int max(int x, int y)
{
    if(x>y) return x;
    else return y;
}
```

# Exemplu goto / return

---

- Exemplu: ieșirea dintr-un ciclu interior direct în afara ciclului exterior.

```
//caută prima apariție a lui b în matricea a
for (i=0;i<n;i++)
    for (j=0;j<n;j++)
        if ( a[i][j]==b ) goto gasit;
printf ("negasit \n");
return;
gasit:
printf("gasit in linia %d si coloana %d \n", i, j);
```

# Probleme propuse

---

1. Să se calculeze și să se afișeze suma:  $S=1+1*2+1*2*3+..+n!$
2. Să se calculeze și să se afișeze suma cifrelor unui număr natural  $n$ .
3. Să se calculeze și să se afișeze inversul unui număr natural  $n$ .
4. Să se afișeze dacă un număr natural dat  $x$  este prim.
5. Să se afișeze primele  $n$  numere naturale prime.
6. Program pentru numărarea și afișarea numerelor prime mai mici ca un întreg dat  $n$ .
7. Să se descompună în factori primi un număr dat  $n$ .
8. Să se afișeze toate numerele naturale mai mici decât 10000 care se pot descompune în două moduri diferite ca sumă de două cuburi.
9. Să se determine elementul maxim, respectiv minim dintr-un șir de  $n$  numere întregi introduse de la tastatură.
10. Să se determine numărul de zile corespunzător unei luni dintr-un anumit an. Luna și anul se vor citi de la tastatură (luna sub forma de întreg).

# Rezolvări

---

```
// determina nr de zile dintr-o lună a unui an nebisect
switch (luna) {
    // februarie
    case 2: zile=28; break;
    // aprilie, iunie,..., noiembrie
    case 4: case 6: case 9: case 11: zile =30; break;
    // ianuarie, martie, mai,.. decembrie
    default: zile=31; break;
}
```