

**Consultații pentru elevii de liceu  
organizate de Facultatea de Matematică și Informatică  
pentru pregătirea concursului Mate-Info și concursului de admitere  
2021**

14.11.2020, 13.00 – 14.15

Lect. Dr. Adriana COROIU

**Algoritmi care lucrează pe numere (fără tablouri sau alte elemente structurate)  
partea I**

**Problema**

**Enunț:**

Sa se determine al n-lea numar din sirul lui Fibonacci.

**Exemple:**

Pentru  $n=3 \Rightarrow$  se va afisa 2

Pentru  $n=5 \Rightarrow$  se va afisa 5

Pentru  $n=7 \Rightarrow$  se va afisa 13

Pentru  $n=9 \Rightarrow$  se va afisa 34

Pentru  $n=10 \Rightarrow$  se va afisa 55

Pentru  $n=11 \Rightarrow$  se va afisa 89

$$\begin{aligned} & n, & n \leq 1 \\ \text{functie1}(n) = & \text{functie1}(n-1) + \text{functie1}(n-2), & n > 1 \end{aligned}$$

Varianta 1 de rezolvare – *varianta recursiva – implementare C++*

```

#include <iostream>

/* crearea functiei pentru determinarea celui al n-lea numar din sirul lui
Fibonacci
    Date de intrare: numarul n
    Date de iesire: numarul din sir corespunzator
*/

int functiei(int n)
{
    if (n <= 1)
        return n;
    return functiei(n-1) + functiei(n-2);
}

int main()
{
    int n;
    std::cout <<"introduceti numarul n=";
    std::cin >>n;

    std::cout <<"Rezultat= "<<functiei(n)<<std::endl;
    return 0;
}

```

### Varianta 1 de rezolvare – *varianta recursiva – implementare Pascal*

```

program problema1;
var numar:integer;
{ crearea functiei pentru determinarea celui al n-lea numar din sirul lui
Fibonacci
    Date de intrare: numarul n
    Date de iesire: numarul din sir corespunzator
}

function functiei(n: integer ):integer;
begin
    if (n <= 1) then functiei := n
    else functiei := functiei(n-1) + functiei(n-2);

end;
begin
write('Introduceti numarul n=');
readln(numar);
writeln( 'Rezultatul este: ', functiei(numar));
end.

```

## Varianta 2 de rezolvare – *varianta iterativa* – *implementare C++*

```
#include <iostream>

/* crearea functiei pentru determinarea celui al n-lea numar din sirul lui
Fibonacci
    Date de intrare: numarul n
    Date de iesire: numarul din sir corespunzator
*/

int functie2(int n)
{
    int a = 0, b = 1, c, i;
    if( n == 0)
        return a;
    for (i = 2; i <= n; i++)
    {
        c = a + b;
        a = b;
        b = c;
    }
    return b;
}

int main()
{
    int n;
    std::cout <<"introduceti numarul n=";
    std::cin >>n;

    std::cout <<"Rezultat= "<<functie2(n)<<std::endl;
    return 0;
}
```

## Varianta 2 de rezolvare – *varianta iterativa – implementare Pascal*

```
program problemaliterativ;
var numar:integer;

{ crearea functiei pentru determinarea celui al n-lea numar din sirul lui
Fibonacci
  Date de intrare: numarul n
  Date de iesire: numarul din sir corespunzator
}

function functie2(n: integer ):integer;
var a,b,c,i:integer;
begin
  a:=0;
  b:=1;

  if( n = 0) then functie2 :=a
  else

    for i:= 2 to n do
      begin
        c := a + b;
        a := b;
        b := c;
      end;
    functie2:=b;
  end;

begin
write('Introduceti numarul n=');
readln(numar);
writeln( 'Rezultatul este: ', functie2(numar));
end.
```

## Conversii intre baze de numeratie

baze de numeratie:

baza 10 : 0 -> 9

baza 2 : 0 si 1

baza 16 : 0 -> 9, A, B, C, D, E, F

$$(77)_{10} = (1001101)_2 = (4D)_{16}$$

$  \begin{array}{rcl}  77 : 2 & = & 38 \text{ r } 1 \uparrow \\  38 : 2 & = & 19 \text{ r } 0 \\  19 : 2 & = & 9 \text{ r } 1 \\  9 : 2 & = & 4 \text{ r } 1 \\  4 : 2 & = & 2 \text{ r } 0 \\  2 : 2 & = & 1 \text{ r } 0 \\  1 : 2 & = & 0 \text{ r } 1  \end{array}  $	$  \begin{array}{rcl}  77 : 16 & = & 4 \text{ r } 13 \uparrow \\  4 : 16 & = & 0 \text{ r } 4  \end{array}  $
---	---

$$(11010)_2 = (26)_{10}$$

$$43210 = 1 \cdot 2^4 + 1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 = 26$$

$$(1A)_{16} = 1 \cdot 16^1 + A \cdot 16^0 = 16 + 10 = 26$$

$$(2AB7)_{16} = 2 \cdot 16^3 + A \cdot 16^2 + B \cdot 16^1 + 7 \cdot 16^0 = (10935)_{10}$$

$$(110011011)_2 = (411)_{10}$$

b2 -> b16

b 16 -> b2

Conversii rapide intre bazele 2 si 16

1 cifra hexazecimala = 1 grup de 4 cifre binare

Base 10	Base 16	Base 2
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
10	A	1010
11	B	1011
12	C	1100
13	D	1101
14	E	1110
15	F	1111

$$0(0010110)_2 = (16)_{16}$$

$$0(010)_2 = (2)_{16}$$

$$(1A2B)_{16} = (0001\ 1010\ 0010\ 1011)_2$$

$$(7EC)_{16} = (0111\ 1110\ 1100)_2$$

Intrebari grila pentru conversii:

1. Care este valoarea in binar a numarului 101 din baza 10?

a. 1110 0101

b. 0110 0101

c. 0101 1010

d. 1010 0101

2. Care este valoarea numarului (0AB34)<sub>h</sub> in binar?

a. 0101 1010 1011 0011 0100

b. 1111.1010 1011 0011 0100

c. 1010 1011 0011 0100 0000

d. 0000 1010 1011 0011 0100

3. Care este valoarea numarului (11 0101)<sub>b</sub> in baza 10?

a. 53

b. 52

c. 54

d. 35

4. Care este valoarea numarului (11 0101)<sub>b</sub> in baza 16?

a. 35

b. 36

c. 34

d. 53

### Problema

#### Enunt:

Se da ecuatia de gradul 2 de forma:  $X^2 - s * X + p = 0$ , cu  $s, p$  apartinand lui  $R$  si  $n$  apartinand lui  $N$ .

Sa se calculeze, in mod recursiv suma puterilor radacinilor  $S_n = X_1^n + X_2^n$ , fara a se calcula radacinile ecuatiei  $X_1$  si  $X_2$ .

#### Exemple:

Daca avem ecuatia:  $X^2 - 6 * X + 8 = 0$  si  $n=2 \Rightarrow$  rezultatul este 20 ( $x_1=4$  si  $x_2=2$ )

Daca avem ecuatia:  $X^2 - 8 * X + 15 = 0$  si  $n=2 \Rightarrow$  rezultatul este 34 ( $x_1=-3$  si  $x_2=-5$ )

#### Explicatii:

Stiind ca  $X_1$  si  $X_2$  sunt radacinile ecuatiei, rezulta relatiile:

$$X_1^2 - s * X_1 + p = 0$$

$$X_2^2 - s * X_2 + p = 0$$

Daca vom inmulti prima relatie cu  $X_1^n$  si pe cea de a doua cu  $X_2^n$  vom obtine:

$$X_1^{n+2} - s * X_1^{n+1} + p * X_1^n = 0$$

$$X_2^{n+2} - s * X_2^{n+1} + p * X_2^n = 0$$

Daca insumam cele doua relatii, vom obtine:

$$(X_1^{n+2} + X_2^{n+2}) - s(X_1^{n+1} + X_2^{n+1}) + p(X_1^n + X_2^n) = 0$$

Ceea ce am putea scrie in mod echivalent:

$$S_{n+2} - s * S_{n+1} + p * S_n = 0 \text{ rezulta deci:}$$

$$S_{n+2} = s * S_{n+1} - p * S_n$$



noi trebuie sa calculam:  $S_n = X_1^n + X_2^n$

$$\Rightarrow S_n = s \cdot S_{n-1} - p \cdot S_{n-2}$$

$$\Rightarrow S_n = X_1^n + X_2^n = s \cdot S_{n-1} - p \cdot S_{n-2}$$

Vom obtine de aici urmatoarea relatie de recurenta pentru suma puterilor radacinilor unei ecuatii de gradul II:

$$Sum(n) = \begin{cases} 2, & \text{daca } n = 0 \\ s, & \text{daca } n = 1 \\ s * Sum(n-1) - p * Sum(n-2), & \text{daca } n > 1 \end{cases}$$

Rezolvare – implementare C++

```
#include <iostream>
float s,p;

/*crearea functiei pentru calculul sumei conform formulei de recurenta
   Date de intrare: numarul n
   Date de iesire: rezultatul sumei conform formulei de recurenta
*/

float Sum (int n)
{
    if (!n) return 2;
    if (n==1) return s;
    return s*Sum(n-1)-p*Sum(n-2);
}

int main()
{
    int n;
    std::cout <<"introduceti cei doi coeficienti: ";
    std::cin >>s>>p;
    std::cout << "n=";
    std::cin >>n;
    std::cout <<"Rezultatul este: "<<Sum(n)<<std::endl;
    return 0;
}
```

Rezolvare – implementare Pascal

```

program problema3;
var numar,s,p:integer;

{crearea functiei pentru calculul sumei conform formulei de recurenta
  Date de intrare: numarul n
  Date de iesire: rezultatul sumei conform formulei de recurenta
}
function suma(n: integer ):integer;
begin
  if n<=0 then suma:=2
  else
    if n=1 then suma:=s
    else suma:=s*suma(n-1)-p*suma(n-2);
  end;

begin
write('Introduceti numarul n=');
readln(numar);

write('Introduceti coeficientul s=');
readln(s);

write('Introduceti coeficientul p=');
readln(p);

writeln( 'Rezultatul este: ', suma(numar));
end.

```

### Probleme tip grilă

1.

a) Variabila x este de tip real. Care dintre următoarele expresii are valoarea 1 dacă și numai dacă numărul real memorat în variabila x aparține intervalului (5,8]?

- a.  $(x < 8) \ \&\& \ (x \geq 5)$
- b.  $(x \leq 8) \ || \ (x > 5)$
- c.  $(x > 8) \ || \ (x \leq 5)$
- d.  $(x \leq 8) \ \&\& \ (x > 5)$

b) Variabilele x și y sunt tip int. Care dintre expresiile de mai jos are valoarea 1 dacă și numai dacă valorile întregi nenule memorate în variabilele x și y sunt egale?

- a.  $(x \% y == 0) \ \&\& \ (y \% x == 0) \ \&\& \ (x * y > 0)$
- b.  $(x \leq y) \ \&\& \ (y < x)$
- c.  $(x \leq y) \ || \ (y \leq x)$
- d.  $x * x == y * y$

2. Se da urmatoarea secventa de cod scrisa in pseudocod.

a)

**citește n** (număr natural)

$z \leftarrow 0$

$p \leftarrow 1$

┌cât timp  $n > 0$  execută

|  $c \leftarrow n \% 10$

|  $n \leftarrow [n/10]$

| ┌dacă  $c \% 3 = 0$  atunci

| |  $z \leftarrow z + p * (9 - c)$

| |  $p \leftarrow p * 10$

| └■

└■

**scrie z**

Care este rezultatul daca se executa pentru i)  $n = 103?$  ii)  $n = 103456?$

a. 86, 863

**b. 96, 963**

c. 76, 763

d. 96, 936

b)

**citește x** (număr natural nenul)

┌cât timp  $x > 0$  execută

| **citește y** (număr natural)

| ┌dacă  $x > y$  atunci

| | **scrie**  $x \% 10$

| | altfel

| | **scrie**  $y \% 10$

| └■

`x ← y`  
■

Care se va afisa dupa executia codului daca se citesc valorile 17 22 13 101 2 7 5 0?

a. 2211775

b. 7231275

c. 1211000

d. 5771122

c. Care este valoarea care se va returna la apelul: `rez = C (12345);`

`int C (int n)`

{

`return ( n / 1000 ) * 100 + n % 100 ;`

}

a. 1245

b. 145

c. 1345

d. 12045

e. 14500

*Raspunsuri grile:*

$1a - d; 1b - a; 2a - b; 2b - a; 2c - a$

## Problema

a) Sa se scrie un subprogram care prin intermediul a trei parametri, notați a, b și c, reprezentand trei valori naturale nenule, fiecare de maximum patru cifre, va returna valoarea 1 dacă cele trei valori ale celor trei parametrii pot constitui laturile unui triunghi și valoarea 0 în caz contrar.

/\*

Descriere: Returneaza valoarea 1 daca valorile parametrilor pot forma un triunghi, 0 in caz contrar.

Date: a, b, c – numare naturale

Rezultate: 1 sau 0

\*/

**Varianta C++**

```
int verificare (int a, int b, int c)
{
    return ((a + b > c) && (a + c > b) && (b + c > a));
}
```

#### Varianta Pascal

```
function verificare(a,b,c:integer):integer;
begin
    if(a+b>c) and (a+c>b) and (b+c>a)
        then verificare:=1
        else verificare:=0;
end;
```

b) Sa se scrie un program care citește de la tastatură șase valori naturale nenule, apoi verifică, utilizând apeluri utile ale subprogramului de la subpunctul anterior dacă primele trei numere citite pot constitui laturile unui triunghi și dacă ultimele trei numere citite pot constitui laturile unui triunghi.

În caz afirmativ, programul va afișa pe ecran mesajul **congruente** dacă cele două triunghiuri sunt congruente sau mesajul **necongruente** dacă cele două triunghiuri nu sunt congruente.

Dacă cel puțin unul dintre cele două triplete de valori nu pot constitui laturile unui triunghi, programul va afișa pe ecran mesajul **nu se poate forma un triunghi**.

**Analiza.** Triunghiurile sunt congruente dacă există cele două triunghiuri și dacă au laturile corespunzătoare congruente (cazul L.L.L.).

Deci, pentru a verifica vom compara lungimile laturilor în ordine corespunzătoare (cea mai mică din primul triunghi cu cea mai mică din al doilea triunghi, până se verifică toate laturile).

Pentru a simplifica procesul de comparare se recomandă ca laturile să fie ordonate

#### Varianta C++

```

#include <iostream>
using namespace std;
/*
Descriere: Returneaza valoarea 1 daca valorile parametrilor pot forma un triunghi, 0
in caz contrar.
Date: a, b, c – numare naturale
Rezultate: 1 sau 0
*/
int a, b, c, d, e, f;
int verificare (int a, int b, int c)
{
    return ((a + b > c) && (a + c > b) && (b + c > a));
}

/*
Descriere: Se interschimba valorile celor doua variabile transmise ca si parametrii
Date: x, y – numare intregi
Rezultate: valorile interschimbate ale parametrilor initiali
*/
void interschimb (int &x, int &y)
{
    int aux = x;
    x = y;
    y = aux;
}

/*
Descriere: Sortarea a trei valori transmise prin parametrii
Date: x, y, z – numare intregi
Rezultate: valorile parametrilor ordonate
*/
void ordonare (int &x, int &y, int &z)
{
    if (x > y) interschimb (x, y);
    if (x > z) interschimb (x, z);
    if (y > z) interschimb (y, z);
}

```

```

int main ()
{
    cin >> a >> b >> c >> d >> e >> f;
    if (verificare (a, b, c) && verificare (d, e, f))
    {
        ordonare (a, b, c);
        ordonare (d, e, f);
        if (a == d && b == e && c == f)
            cout << "triunghiuri formate congruente";
        else
            cout << "triunghiuri formate necongruente";
    }
    else
        cout << "NU se poate forma un triunghi";
}

```

### Varianta Pascal

```

var a,b,c,d,e,f:integer;

```

```

{

```

Descriere: Returneaza valoarea 1 daca valorile parametrilor pot forma un triunghi, 0 in caz contrar.

Date: a, b, c – numare naturale

Rezultate: 1 sau 0

```

}

```

```

function verificare(a,b,c:integer):integer;

```

```

begin

```

```

    if(a+b>c) and (a+c>b) and (b+c>a)

```

```

        then verificare:=1

```

```

        else verificare:=0;

```

```

end;

```

```

{

```

Descriere: Se interschimba valorile celor doua variabile transmise ca si parametrii

Date: x, y – numare intregi

Rezultate: valorile interschimbate ale parametrilor initiali

```

}

```

```
procedure interschimb(var x,y:integer);
```

```
var aux:integer;
```

```
begin
```

```
    aux:=x;
```

```
    x:=y;
```

```
    y:=aux;
```

```
end;
```

```
{
```

Descriere: Sortarea a trei valori transmise prin parametrii

Date: x, y, z – numare intregi

Rezultate: valorile parametrilor ordonate

```
}
```

```
procedure ordonare(var x,y,z:integer);
```

```
begin
```

```
    if(x>y) then interschimb(x,y);
```

```
    if(x>z) then interschimb(x,z);
```

```
    if(y>z) then interschimb(y,z);
```

```
end;
```

```
begin
```

```
    readln(a,b,c,d,e,f);
```

```
    if(verificare(a,b,c)=1) and (verificare(d,e,f)=1)
```

```
        then
```

```
            begin
```

```
                ordonare(a,b,c);
```

```
                ordonare(d,e,f);
```

```
                if(a=d) and (b=e) and (c=f)
```

```
                    then
```

```
                        writeln('triunghiuri formate congruente')
```

```
                    else
```

```
                        writeln('triunghiuri formate necongruente');
```

```
            end
```

```
        else
```

```
            writeln('NU se poate forma un triunghi');
```

```
end.
```