

Laborator 9

Stream-urile cin și cout

În acest capitol sunt prezentate considerații teoretice și probleme rezolvate privind definirea și utilizarea stream-urilor cin și cout în C++.

CONSIDERAȚII TEORETICE

Stream-urile cin și cout din <iostream.h> sunt utilizate în C++ în locul funcțiilor de citire/scriere utilizate în C. Semnificația lor este :

- cin (“see in”) = input stream
- cout (“see out”) = output stream

cin și cout se utilizează cu operatorii << și >> care au următoarea semnificație:

- << (“put to”) = operator de inserție
- >> (“get from”) = operator de extracție

Formatul de utilizare al acestor streamuri este :

cin >> var1 >> var2 ...;
cout << var1 << var2 ...;

unde **var1**, **var2**, ... pot fi numai nume de variabile în cazul lui **cin** nu și constante sau expresii ca în cazul lui **cout**

Efectul operației de inserție este descris pentru tipurile standard char, int și float în tabelul de mai jos:

Tipul variabilei din operația de inserție cin	Data considerată validă
char	Un singur caracter diferit de blank (spațiu)
int	O constantă întreagă opțional precedată de un semn
float	O constantă întreagă sau reală (eventual specificată în format științific-utilizând notația cu e), opțional precedată de semn

Ex. : citirea unui întreg

```
int i;
```

```
cin >> i;
```

Efect: similar cu efectul instrucțiunilor:

```
int i;
```

```
scanf(“%d”, &i);
```

Exemple de utilizare a streamului cin:

Instrucțiune de inserție	Data intrare	Conținut variabilă după inserție
1. cin >> i;	32	i=32
2. cin >> i >> j;	4 60	i=4, j=60
3. cin >> i >> ch >> x;	25 A 16.9	i=25, ch='A', x=16.9

4. cin >> i >> ch >> x;	25 A 16.9	i=25, ch='A', x=16.9
5. cin >> i >> ch >> x;	25A16.9	i=25, ch='A', x=16.9
6. cin >> i >> j >> x;	12 8	i=12, j=8 –calculatorul așteaptă introducerea lui x
7. cin >> i >> x;	46 32.4 15	i=46,x=32.4 –calculatorul reține ultimul nr. (15) pentru o inserție ulterioară; dacă aceasta nu se realizează aceste date se pierd.

Caracterul linie nouă : “\n”(new line) și endl se utilizează pentru trecerea pe linie nouă. În exemplele de mai jos se ilustrează modul de utilizare a streamului cout.

Ex.: operații de scriere (afisare) echivalente

```
cout << "Hello \n";
cout << "Hello" << endl;
```

Ex. : scrierea (afișarea) unui text

```
cout << "Hello student !\n";
```

Efect: similar cu efectul instrucțiunii:

```
printf("Hello student !\n");
```

Funcția **cin.get()** se utilizează pentru citirea datelor de tip caracter și are următorul format de apelare: **cin.get(caracter)**; Se preia caracterul introdus de la tastatură indiferent dacă acesta este spațiu sau linie nouă ('\n')

Diferența dintre cin și cin.get(): cin ignoră spațiile cin.get() preia și spațiile sau caracterul de linie nouă introduse de la tastatură.

Ex. :

```
cin >>ch1 >>ch2;
```

Dacă se introduce de la tastatură:

```
>R 1
```

atunci lui ch1 i se atribuie caracterul “R” se sare peste spațiu și se atribuie lui ch2 , caracterul “1”
Atenție: caracterul “1” este interpretat de către calculator în mod diferit de întregul “1”!

Dacă se dorește și citirea spațiului dintre R și 1 atunci se utilizează cin.get()

```
cin .get(ch1);
cin .get(ch2);
cin .get(ch3);
```

Funcția ignore() se utilizează pentru ignorarea unui număr de date citite și are următorul format de apelare:

```
cin .ignore(intreg, caracter);
```

unde **intreg** reprezintă un nr. intreg sau o expresie de tip întreg
caracter este o constantă de tip caracter

Ex.: ignorarea primelor 200 de caractere introduse de la tastatura sau ignorarea tuturor caracterelor până se introduce caracterul de linie nouă. Se realizează condiția care este îndeplinită prima !

```
cin.ignore(200, '\n');
```

Ex.: se ignoră primele 100 de caractere introduse de la tastatură sau se ignoră toate caracterele până se întâlnește litera "B" : cin.ignore(100, 'B');

Ex.: se ignoră primele 2 caractere introduse de la tastatură : cin.ignore(2, '\n');

Probleme rezolvate

Ex.1 : Programul C++ calculează pătratul și cubul unui nr. întreg utilizând cin/cout.

Varianta in C++
<pre>#include <iostream> #include <math.h> using namespace std; int Patrat(int n); int Cub(int n); int main() {int a, b; cout << "a= "; cin >> a ; cout << "patratul lui " << a << " este " << Patrat(a) << endl; cout << "iar cubul lui " << a << " este " << Cub(a) << endl; cout << "b="; cin >> b; cout << a << " la puterea " << b << " este " << pow(a,b) << endl; return 0; } int Patrat(int n) { return n * n;} int Cub(int n) { return n * n * n; }</pre>

Rezultate:

```
a= 2
patratul lui 2 este 4
iar cubul lui 2 este 8
b=10
2 la puterea 10 este 1024
```

Aplicație:

Scrieți programul care calculează și afișează valorile funcției

$$f(x) = \begin{cases} 3x^2 + 2x - 1, & x < 0 \\ 5x + 2, & x \geq 0 \end{cases}, \text{ pentru orice}$$

valoare x introdusă de la tastatură cu cin

Ex.2. Programul C++ calculează și afișează valoarea ipotenuzei unui triunghi dreptunghic utilizând cin și cout.

Varianta in C++
<pre>#include <iostream> #include <math.h> using namespace std; int main() { float catetaA, catetaB, ipoten; cout << "Introduceti catetele: " << endl;</pre>

```

cout << "cateta A="; cin >> catetaA ;
cout << "cateta B=" ;cin >> catetaB ;
ipoten = sqrt(catetaA * catetaA+ catetaB*catetaB);
cout << "Ipotenuza este: " << ipoten << endl;
return 0;}

```

Rezultate:

```

Introduceti catetele:
cateta A=50
cateta B=60
Ipotenuza este: 78.1025

```

Aplicație:

Scrieți programul care rezolvă ec. de gradul I cu cin, cout

Ex.3 : Programul C++ calculează pătratul unui nr. întreg utilizând o funcție recursivă, și cin , cout.

Varianta in C++

```

#include <iostream>
using namespace std;
int Power( int, int );
int main()
{ int number;      // Numarul intreg
  int exponent;    // Exponentul intreg
  cout<<"numar=";cin >> number ;
  cout<<"\nexponent="; cin >> exponent;
  cout << "numar ^ exponent="<<Power(number, exponent);
return 0;}
int Power( int x, int n ) // functia de ridicare la putere
// Presupunem : n > 0
// Atentie la depasirea intervalului int pentru exponent prea mare
{ if (n == 1) return x;
  else return x * Power(x, n - 1); // apel recursiv
}

```

Rezultate:

```

numar=2
exponent=10
numar ^ exponent=1024

```

Aplicație:

Modificați programul astfel încât să calculați și afișați n! utilizând cin, cout

Ex.4 : Programul C++ citește valoarea temperaturii (t) de afară și afișează o activitate recomandată conform tabelului de mai jos:

Temperatură	Activitate
$t > 35^{\circ} C$	inot
$35 > t > 25$	Tenis
$25 > t > 20$	golf
$20 > t > 0$	schi
$t < 0^{\circ} C$	discoteca

Varianta in C++

```

#include <iostream>

```

```
using namespace std;
int main()
{ int temperature;
//preluarea temperaturii
cout << "Introduceti temperatura in grade C:" << endl;
cin >> temperature;
cout << "Temperatura este : " << temperature << " grade C." << endl;
// Afisarea activitatii recomandate
cout << "La aceasta temperatura, va recomandam: ";
if (temperature > 35) cout << "inot." << endl;
else if (temperature > 25) cout << "tenis." << endl;
else if (temperature > 20) cout << "golf." << endl;
else if (temperature < 0) cout << "schi." << endl;
else cout << "discoteca." << endl;
return 0;}
```

Rezultate:

```
Introduceti temperatura in grade C:
25
Temperatura este : 25 grade C.
La aceasta temperatura, va recomandam: golf.
```

Aplicație:

Modificați programul astfel încât să calculați și afișați temperatura în grade Fahrenheit: $[^{\circ}\text{C}] = ([^{\circ}\text{F}] - 32) \times 5/9$.

Ex.5. Programul C++ calculează pentru orice x real, valoarea funcției f(x) definită prin expresia:

$$f(x) = \begin{cases} 3x^2 + 2x - 10, & x < 0 \\ x^2, & x > 0 \\ 2, & x = 0 \end{cases}$$

Varianta in C++

```
#include <iostream>
using namespace std;
int main(void)
{float x,f;
cout<<"introduceti valoarea lui x:"<<cin >>x;
if (x<0.) f=3.0*x*x+2.0*x-10.0;
else if (x==0.) f=2.0;
else f=x*x;
cout <<"f(x)=" << f <<endl;
return 0;}
```

Rezultate:

```
introduceti valoarea lui x:10.5
f(x)=110.25
```

Aplicație:

Modificați programul astfel încât să calculați și afișați valoarea funcției $f(x) = x^3 - 1$.

Ex.6. Programul C++ calculează și afișează valorile primilor n termeni ai șirului: $a_i = \left(1 + \frac{1}{i}\right)^i$, calculându-se pentru fiecare din valorile lui a_i și abaterile acestora de la valoarea exactă a nr. $e=2.7182818285$. Problema se bazează pe următoarele limite cunoscute: $e = \lim_{n \rightarrow \infty} \left(1 + \frac{1}{n}\right)^n$ și

$$e = \lim_{n \rightarrow \infty} \frac{n}{\sqrt[n]{n!}}$$

Varianta in C++

```
#include <iostream>
#include <math.h>
using namespace std;
int main (void)
{int i, n; double e,a;
cout << "n=";cin >>n;
e=exp(1);
cout << "e=" << e << endl;
for (i=1;i<=n;i++)
{a= pow((1.+1./i),i) ;
cout << endl << a << "\t" << ", abatere: " << exp(1)-a ; }
cout << endl;
return 0;}
```

Rezultate:

```
n=10
e=2.71828
2          , abatere: 0.718282
2.25      , abatere: 0.468282
2.37037   , abatere: 0.347911
2.44141   , abatere: 0.276876
2.48832   , abatere: 0.229962
2.52163   , abatere: 0.196655
2.5465    , abatere: 0.171782
2.56578   , abatere: 0.152497
2.58117   , abatere: 0.137107
2.59374   , abatere: 0.124539
```

Ex.7. Programul C++ calculează și afișează rădăcinile ecuației de gradul II: ax^2+bx+c , pentru a,b,c numere reale introduse de la tastatură. Se vor utiliza 2 funcții: *citire()*-pentru citirea coeficienților ecuației și *rezolv()*-pentru rezolvarea ecuației.

Varianta in C++

```
#include <iostream>
#include <math.h>
using namespace std;
double a,b,c,y,y2,x1,x2,D,x;
char i;
void citire()
{ cout <<"Fie ecuatia de gradul 2: aX*X + bX + c = 0" << endl;
cout << endl <<"Introduceti coeficientii ecuatiei: " <<endl;
cout << endl << "a=";cin >> a;
cout << "b="; cin >> b;
cout << "c=";cin >> c;}
void rezolv()
{ if(a==0) { cout << endl <<"Ecuatia este de gradul I";
            if(b!=0) {x=-c/b;
                    cout << endl <<"Solutia ecuatiei este: " << x;}
            else cout <<"Eroare. Introduceti din nou coeficientii";
        }
    else
        {cout << "Ecuatia este " << a <<"X*X +"<<b <<"X +" << c;
```

```

    cout << "= 0" << endl;
D=b*b-4*a*c;
cout << endl <<"Discriminantul ecuatiei:" <<D;
if(D>=0) { x1=(-b+sqrt(D))/(2.*a);
          x2=(-b-sqrt(D))/(2.*a);
          cout <<endl << "Ecuatia are radacini reale: x1 = " << x1 <<" , x2 = " <<x2;
        }
else { x1=-b/(a*2.);y=sqrt(-D)/(2.*a);
      x2=-b/(a*2.);y2=sqrt(-D)/(2.*a);
      cout << endl << "Ecuatia are radacini complexe:" << endl;
      cout << "z1 = " << x1 << " + " <<y <<"*i, z2 = " << x2 << " - " << y2;
      cout << "*i" << endl;}
}}
int main()
{citire();
rezolv();
return 0;}

```

Rezultate:

```

Fie ecuatia de gradul 2: aX*X + bX + c = 0
Introduceti coeficientii ecuatiei:
a=2
b=5.5
c=-1.4
Ecuatia este 2X*X +5.5X +-1.4= 0
Discriminantul ecuatiei:41.45
Ecuatia are radacini reale: x1 = 0.234542, x2 = -2.98454

```

Ex.8. Programul C++ determină și afișează n termeni ai șirului lui Fibonacci utilizând 3 metode diferite:

a) Metoda recursivă ținând cont de relațiile: $f_k=f_{k-1}+f_{k-2}$, $f_0=0$, $f_1=1$, $k \geq 2$

b) O metodă nerecursivă

c) O altă metodă nerecursivă ținând cont de relațiile: $f_k = \frac{1}{\sqrt{5}} \left[\left(\frac{1+\sqrt{5}}{2} \right)^k - \left(\frac{1-\sqrt{5}}{2} \right)^k \right]$,

$$\Phi^{k-2} \leq f_k \leq \Phi^{k-1}$$

$$\Phi = \frac{1+\sqrt{5}}{2} \approx 1.6180339887 = \text{raportul de aur}$$

Varianta in C++

```
#include <iostream>
#include <math.h>
using namespace std;
int n, k;
int main ()
{int n, a=1,b=0,c,f1[100], f2[100],f3[100];
double fi1, fi2;
f1[0]=f2[0]=f3[0]=0;
f1[1]=f2[1]=f3[1]=1;
fi1=(1+sqrt(5))/2;
fi2=(1-sqrt(5))/2;
cout <<"Introduceti n>=0, n=";
cin >>n;
if (n==0)
{cout <<"f1[ 0]=" << f1[0];
    cout <<"f2[ 0]=" << f2[0];
    cout <<"f3[ 0]=" << f3[0];
}
else if (n==1)
{cout <<"f1[ 1]=" << f1[1];
    cout <<"f2[ 1]=" << f2[1];
    cout <<"f3[ 1]=" << f3[1];
}
for (k=2;k<=n;k++)
//metoda recursiva 1
{f1[k]=f1[k-1]+f1[k-2];
cout <<"f1[ "<< k << "]" << f1[k] << "\t";
//metoda nerecursiva 2
c=a; a+=b;b=c;f2[k]=a ;
cout <<"f2[ "<< k << "]" << f2[k] << "\t";
//metoda nerecursiva 3
f3[k]=round(((1/sqrt(5))*(pow(fi1,k)-pow(fi2,k))));
cout << "f3[ "<< k << "]" << f3[k] << endl ;}
return 0;}
```

Rezultate:

```
Introduceti n>=0, n=15
f1[ 2]=1      f2[ 2]=1      f3[ 2]=0
f1[ 3]=2      f2[ 3]=2      f3[ 3]=2
f1[ 4]=3      f2[ 4]=3      f3[ 4]=3
f1[ 5]=5      f2[ 5]=5      f3[ 5]=5
f1[ 6]=8      f2[ 6]=8      f3[ 6]=8
f1[ 7]=13     f2[ 7]=13     f3[ 7]=13
f1[ 8]=21     f2[ 8]=21     f3[ 8]=21
f1[ 9]=34     f2[ 9]=34     f3[ 9]=34
f1[ 10]=55    f2[ 10]=55    f3[ 10]=55
f1[ 11]=89    f2[ 11]=89    f3[ 11]=89
f1[ 12]=144   f2[ 12]=144   f3[ 12]=144
f1[ 13]=233   f2[ 13]=233   f3[ 13]=233
f1[ 14]=377   f2[ 14]=377   f3[ 14]=377
f1[ 15]=610   f2[ 15]=610   f3[ 15]=610
```

Rezultate:


```

C:\ Visual C\2\Lab10\Debug\lex8.exe
Introduceti n>=0, n=20
f1[ 2]=1      f2[ 2]=1      f3[ 2]=1
f1[ 3]=2      f2[ 3]=2      f3[ 3]=2
f1[ 4]=3      f2[ 4]=3      f3[ 4]=3
f1[ 5]=5      f2[ 5]=5      f3[ 5]=5
f1[ 6]=8      f2[ 6]=8      f3[ 6]=8
f1[ 7]=13     f2[ 7]=13     f3[ 7]=13
f1[ 8]=21     f2[ 8]=21     f3[ 8]=21
f1[ 9]=34     f2[ 9]=34     f3[ 9]=34
f1[ 10]=55    f2[ 10]=55    f3[ 10]=55
f1[ 11]=89    f2[ 11]=89    f3[ 11]=89
f1[ 12]=144   f2[ 12]=144   f3[ 12]=144
f1[ 13]=233   f2[ 13]=233   f3[ 13]=233
f1[ 14]=377   f2[ 14]=377   f3[ 14]=377
f1[ 15]=610   f2[ 15]=610   f3[ 15]=610
f1[ 16]=987   f2[ 16]=987   f3[ 16]=987
f1[ 17]=1597  f2[ 17]=1597  f3[ 17]=1597
f1[ 18]=2584  f2[ 18]=2584  f3[ 18]=2584
f1[ 19]=4181  f2[ 19]=4181  f3[ 19]=4181
f1[ 20]=6765  f2[ 20]=6765  f3[ 20]=6765
Press any key to continue

```

PROBLEME PROPUSE:

1. Să se scrie un program C++ utilizând cin și cout care să citească un șir de la tastatură și să:
 - determine și afișeze, elementele minim și maxim ,
 - calculeze și afișeze suma și produsul elementelor
 - ordoneze șirul prin metoda bulelor

2. Să se scrie un program C++ utilizând cin și cout care să citească un șir de caractere și să:
 - afișeze șirul invers,
 - numere vocalele și consoanele din șir
 - afișeze toate literele cu majuscule

3. Să se scrie un program C++ care citește de la tastatură un număr n întreg și valorile reale ale n rezistențe și calculează și afișează valoarea rezistenței echivalente serie și paralel.

4. Să se scrie un program C++, utilizând cin și cout, în care se definește o variabilă tablou de structuri numită angajați de tip structură cu câmpurile: nume, adresă, cod numeric, salar net și inițializați acest tablou cu n articole de acest tip (n introdus de la tastatură). Să se afișeze numai angajații care au salarul cuprins între 400 și 1.000 RON.

5. Să se scrie un program C++, utilizând cin și cout, care realizează următoarele operații:
 - citește de la tastatură un număr întreg reprezentând studenții unei grupe, numele, prenumele și 2 note pentru fiecare student
 - calculează media aritmetică a notelor pentru fiecare student
 - afișează studenții sortați în ordinea mediilor.