

Laborator nr. 8

Tablouri unidimensionale – vectori (partea II)

A. Probleme rezolvate:

Problema 1: Se dă un vector cu n elemente, numere naturale. Afișați în ordine crescătoare valorile prime din acest vector.

Date de intrare: Programul citește de la tastatură numărul n, iar apoi n numere naturale, reprezentând elementele vectorului.

Date de ieșire: Programul va afișa pe ecran valorile prime din vector, în ordine crescătoare, separate prin exact un spațiu.

Exemplu

Intrare

7

13 1 10 15 3 7 11

Ieșire

3 7 11 13

Soluție:

```
#include <iostream>
using namespace std;
int a[1001],i,j,n,b,m;
int main()
{
    cin>>n;
    for(i = 1; i<= n; i++)
    {
        cin>>b;
        int k = 1;
        for(j = 2; j * j <= b && k; j++)
            if(b % j == 0) k = 0;
        if(k && b > 1) a[++m] = b;
    }
    for(i = 1; i < m; i++)
        for(j = i + 1; j <= m; j++)
            if(a[i] > a[j])
                swap(a[i], a[j]);
    for(i = 1; i <= m; i++) cout<<a[i]<<" ";
    return 0;
}
```

Laborator - Programarea Calculatoarelor si Limbaje de Programare (2022) Limbajul C++

Adrian Runceanu

Soluție implementată în compilatorul online <https://www.jdoodle.com/online-compiler-c++/>:

```
1 #include <iostream>
2 using namespace std;
3 int a[1001],i,j,n,b,m;
4 int main()
5 {
6     cin>>n;
7     for(i = 1; i <= n; i++)
8     {
9         cin>>b;
10        int k = 1;
11        for(j = 2; j * j <= b && k; j++)
12            if(b % j == 0)
13                k = 0;
14        if(k && b > 1) a[++] = b;
15    }
16    for(i = 1; i < m; i++)
17        for(j = i + 1; j <= m; j++)
18            if(a[i] > a[j])
19                swap(a[i], a[j]);
20    for(i = 1; i <= m; i++)
21        cout<<a[i]<<" ";
22    return 0;
23 }
```

Executa Mode, Version, Inputs & Arguments

GCC 9.1.0 Interactive Stdin Inputs

CommandLine Arguments

7
13 1 10 15 3 7 11

Execute

Result
CPU Time: 0.00 sec(s), Memory: 3416 kilobyte(s)

3 7 11 13

Problema 2: Orice șir se încadrează în următoarele categorii: **șir constant**, **șir strict crescător**, **șir crescător**, **șir strict descrescător**, **șir descrescător** sau **șir neordonat**.

Se citește un șir cu n elemente naturale. Să se verifice în ce categorie se încadrează.

Date de intrare: Programul citește de la tastatură numărul n, iar apoi n numere naturale, reprezentând elementele șirului.

Date de ieșire: Programul va afișa pe ecran unul dintre mesajele șir constant, șir strict crescător, șir crescător, șir strict descrescător, șir descrescător sau șir neordonat. Se va tipări categoria cea mai strictă în care se încadrează șirul.

Exemplu 1:

Intrare

5

1 2 3 4 5

Ieșire

șir strict crescător

Exemplu 2:

Intrare
5
5 3 3 2 1
Ieşire
sir descrescator

Exemplu 3:

Intrare
5
1 1 1 1 1
Ieşire
sir constant

Exemplu 4:

Intrare
5
1 2 3 1 5
Ieşire
sir neordonat

Solutie:

```
#include <iostream>
using namespace std;
int x[1001], n, i, a = 1, b = 1, c = 1, d = 1, e = 1;
int main()
{
    cin>>n;
    for(i=1;i<=n;i++)    cin>>x[i];
    for(i=1;i<n;i++){
        if(x[i] != x[i+1])    a=0;
        if(x[i] >= x[i+1])    b=0;
        if(x[i] <= x[i+1])    c=0;
        if(x[i] > x[i+1])    d=0;
        if(x[i] < x[i+1])    e=0;
    }
    if(a) cout<<"sir constant";
    else if(b) cout<<"sir strict crescator";
        else if(c) cout<<"sir strict descrescator";
            else if(d) cout<<"sir crescator";
                else if(e) cout<<"sir descrescator";
                    else cout<<"sir neordonat";
    return 0;
}
```

Soluție implementată în compilatorul online <https://www.jdoodle.com/online-compiler-c++/>:

```
1 #include <iostream>
2 using namespace std;
3 int x[1001],n,i,a=1,b=1,c=1,d=1,e=1;
4 int main()
5 {
6     cin>>n;
7     for(i=1;i<=n;i++)
8         cin>>x[i];
9     for(i=1;i<=n;i++)
10    {
11        if(x[i] != x[i-1]) a=0;
12        if(x[i] >= x[i-1]) b=0;
13        if(x[i] <= x[i-1]) c=0;
14        if(x[i] > x[i-1]) d=0;
15        if(x[i] < x[i-1]) e=0;
16    }
17    if(a) cout<<"sir constant";
18    else
19        if(b) cout<<"sir strict crescator";
20        else
21            if(c) cout<<"sir strict descrescator";
22            else
23                if(d) cout<<"sir crescator";
24                else
25                    if(e) cout<<"sir descrescator";
26                    else
27                        cout<<"sir neordonat";
28    return 0;
29 }
```

Execute Mode, Version, Inputs & Arguments

GCC 9.1.0 Interactive Stdin Inputs

CommandLine Arguments

5
1 2 3 1 5

Execute

Result

CPU Time: 0.00 sec(s), Memory: 3284 kilobyte(s)

```
sir neordonat
```

Problema 3: Se dă un vector cu n numere naturale. Să se determine câte dintre elemente au valoarea strict mai mare decât media aritmetică a elementelor vectorului.

Date de intrare: Programul citește de la tastatură numărul n, iar apoi n numere naturale, reprezentând elementele vectorului.

Date de ieșire: Programul afișează pe ecran numărul C, reprezentând valoarea cerută.

Exemplu

Intrare

5

5 0 2 1 4

Ieșire

2

Explicație: Media aritmetică a elementelor este 2.4. În vector sunt 2 elemente cu valoarea strict mai mare decât 2.4: 5 4

Solutie:

```
#include <iostream>
using namespace std;
int a[201],i,n,s,p;
int main()
{
    cin>>n;
    for(i=1;i<=n;i++) {
        cin>>a[i];
        s += a[i];
    }
    for(i = 1; i <= n; i++)
        if(a[i] > s / n) p++;
    cout<<p;
    return 0;
}
```

Soluție implementată în compilatorul online <https://www.jdoodle.com/online-compiler-c++/>:

The screenshot displays an online C++ compiler interface. At the top, the C++ code is shown in a text editor with line numbers 1 through 16. Below the code editor, there is a section for execution settings, including a dropdown menu for the compiler version (GCC 9.1.0), a checkbox for 'Interactive' mode, and a 'Stdin Inputs' field containing the values '5' and '5 0 2 1 4'. A 'CommandLine Arguments' field is also present but empty. A blue 'Execute' button is located at the bottom right of the settings area. Below the settings, the 'Result' section shows the output '2' on a black background. Performance metrics at the bottom indicate 'CPU Time: 0.00 sec(s), Memory: 3500 kilobyte(s)'.

Problema 4: Se consideră un șir a_1, a_2, \dots, a_n de numere naturale.

- Să se afișeze elementele șirului de la dreapta la stânga.
- Să se calculeze suma valorilor pare din șir.
- Să se determine suma valorilor aflate pe poziții pare în șir.
- Să se determine numărul numerelor din șir care sunt divizibile cu 10.
- Să se determine suma numerelor divizibile cu 3 și aflate pe poziții impare.

Date de intrare: Programul citește de la tastatură numărul n , iar apoi șirul de n numere naturale, separate prin spații.

Date de ieșire: Programul va afișa pe ecran pe prima linie elementele șirului de la dreapta la stânga. Pe linia a doua se va afișa un singur număr reprezentând suma valorilor pare. Pe linia a treia se va afișa un singur număr reprezentând suma valorilor aflate pe poziții pare în șir. Pe linia a patra se va afișa un singur număr reprezentând numărul numerelor din șir care sunt divizibile cu 10. Pe linia a cincea se va afișa un singur număr reprezentând suma numerelor divizibile cu 3 și aflate pe poziții impare.

Exemplu

Intrare

10

1 2 3 4 5 6 7 8 9 10

Ieșire

10 9 8 7 6 5 4 3 2 1

30

30

1

12

Soluție:

```
#include <iostream>
using namespace std;
int x[101],n,i,s,s1,s2,nr;
int main()
{
    cin>>n;
    for(i = 1; i <= n; i++) cin>>x[i];
    for(i = n; i >= 1; i--) cout<<x[i]<<" ";
    cout<<endl;
    for(i = 1; i <= n; i++){
        if(x[i] % 2 == 0) s += x[i];
        if(i % 2 == 0) s1 += x[i];
        if(x[i] % 10 == 0) nr++;
        if(i % 2 != 0 && x[i] % 3 == 0) s2 += x[i];
    }
    cout<<s<<endl<<s1<<endl<<nr<<endl<<s2;
    return 0;
}
```

Laborator - Programarea Calculatoarelor si Limbaje de Programare (2022) Limbajul C++

Adrian Runceanu

Soluție implementată în compilatorul online <https://www.jdoodle.com/online-compiler-c++/>:

```
1 #include <iostream>
2 using namespace std;
3 int x[10],n,s1,s2,nr;
4 int main()
5 {
6     cin>>n;
7     for(i = 1; i <= n; i++) cin>>x[i];
8     for(l = n; l >= 1; l--) cout<<x[l]<<" ";
9     cout<<endl;
10    for(i = 1; i <= n; i++)
11    {
12        if(x[i] % 2 == 0) s += x[i];
13        if(i % 2 == 0) s1 += x[i];
14        if(x[i] % 10 == 0) nr++;
15        if(i % 2 != 0 && x[i] % 3 == 0) s2 += x[i];
16    }
17    cout<<endl<<s1<<endl<<nr<<endl<<s2;
18    return 0;
19 }
```

Executa Mode, Version, Inputs & Arguments

GCC 9.1.0 Interactive Stdin Inputs

CommandLine Arguments

10
1 2 3 4 5 6 7 8 9 10

Execute

Result

CPU Time: 0.00 sec(s), Memory: 3272 kilobyte(s)

```
10 9 8 7 6 5 4 3 2 1
30
30
1
12
```

Problema 5: Se dă un șir cu n elemente, numere naturale. Să se afișeze, pentru fiecare element din șir, valoarea din șir aflată după acesta și mai mare decât acesta (**Următorul Element Mai Mare**). Dacă o asemenea valoare nu există, se va afișa -1.

Date de intrare: Programul citește de la tastatură numărul n, iar apoi cele n elemente ale șirului.

Date de ieșire: Programul va afișa pe ecran cele n valori determinate, separate prin câte un spațiu.

Exemplu

Intrare

5

3 4 3 5 1

Ieșire

4 5 5 -1 -1

Solutie:

```
#include <iostream>
using namespace std;
int x[1001], n, i, j, ok;
int main()
{
    cin>>n;
    for(i = 1; i <= n; i++) cin>>x[i];
    for(i = 1; i < n; i++){
        ok = 0;
        for(j = i + 1; j <= n && !ok; j++)
            if(x[i] < x[j]){
                cout<<x[j]<<" ";
                ok = 1;
            }
        if(!ok) cout<<-1<<" ";
    }
    cout<<-1;
    return 0;
}
```

Soluție implementată în compilatorul online <https://www.jdoodle.com/online-compiler-c++/>:

The screenshot displays an online C++ compiler interface. At the top, the C++ code is shown in a text editor with line numbers 1 through 21. Below the code editor, there is a control panel with a dropdown menu set to 'GCC 9.10', an 'Interactive' checkbox, and a 'Stdin Inputs' field containing the number '5' and the sequence '3 4 3 5 1'. A 'CommandLine Arguments' field is empty. A blue 'Execute' button is visible. Below the control panel, the 'Result' section shows the output: '4 5 5 -1 -1'. At the bottom, the execution statistics are listed: 'CPU Time: 0.00 secs, Memory: 3444 kilobyte(s)'.

B. Probleme propuse spre rezolvare:

L8_1). Se citește un tablou unidimensional cu n ($1 \leq n \leq 100$) componente numere naturale. Se cere să se construiască și să se afișeze un nou vector cu componentele *pătrate perfecte* ale vectorului inițial.

L8_2). Se da un tablou unidimensional cu n ($1 \leq n \leq 100$) componente numere naturale. Să se verifice dacă componentele sunt în ordine crescătoare de la stânga la dreapta.

L8_3). Să se rearanjeze elementele unui tablou unidimensional de numere întregi, astfel încât numerele pare să apară înaintea numerelor impare. În cadrul subsecvenței de numere pare, respectiv impare, elementele trebuie să apară în ordinea în care erau în vectorul inițial.

L8_4). Se consideră un tablou unidimensional x cu n numere întregi. Se cere să se calculeze cel mai mare divizor comun al lor.

L8_5). Se da un tablou unidimensional X cu n numere reale și se cere să se modifice astfel încât să se intercaleze între oricare două elemente consecutive, media lor aritmetică.

Bibliografie

[1] <http://www.pbinfo.ro> Descrierea site-ului: "*www.pbinfo.ro îți propune să rezolvi probleme de informatică, cu evaluator automat. Știi pe loc dacă soluția ta este corectă sau dacă trebuie să mai lucrezi la ea.*

Problemele sunt grupate după programa de informatică pentru liceu. Dar nu trebuie să fii la liceu ca să rezolvi aceste probleme. Poți fi elev de gimnaziu, student, profesor sau pur și simplu pasionat de informatică. De fapt, trebuie doar să vrei!!"

[2] <https://www.runceanu.ro/adrian>

[3] Adrian Runceanu – „Programarea și utilizarea calculatoarelor”, Editura Academica Brâncuși din Târgu-Jiu, 2003, ISBN 973-8436-44-3

[4] Adrian Runceanu, Mihaela Runceanu, „Noțiuni de programare – limbajul C++”, Editura Academica Brâncuși din Târgu-Jiu, 2012, ISBN 978-973-144-550-2

[5] Adrian Runceanu, Mihaela Runceanu - „Algoritmi implementați în limbajul C++. Volumul I – Algoritmi elementari”, Editura Academica Brâncuși din Târgu Jiu, 2021, ISBN 978-606-9614-06-8