

## Laborator 6 - limbajul Java

### Tablouri in limbajul Java - Tablouri unidimensionale (vectori)

#### Probleme rezolvate:

Scriti, compilati si executati toate exemplele din acest laborator:

1. Se citeste un numar n si un vector cu n componente numere intregi. Sa se afiseze continutul componentelor vectorului.

```
import java.util.Scanner;
public class lab_6_1 {
    public static void main(String[] args){

        System.out.println("Primul program in Java care citeste si afiseaza elementele
unui vector");
        int n;
        Scanner input = new Scanner(System.in);
        n = input.nextInt();
        System.out.println("Numarul de elemente = "+n);
        /* declararea variabilei de tip tablou unidimensional V[]*/

        int V[] = new int[n];
        // citirea elementelor in tabloul dat
        int i;
        for(i=0;i<n;i++)  V[i] = input.nextInt();

        // afisarea elementelor tabloului
        System.out.print("Vectorul citit este: ");
        for(i=0;i<n;i++)  System.out.print(" "+V[i]);
        }
}
```

## Laborator - Programare orientată pe obiecte Limbajul Java

Adrian Runceanu

```

1  /*
2  1. Se citeste un numar n si un vector cu n componente numere intregi.
3  Sa se afiseze continutul componentelor vectorului.
4  */
5  import java.util.Scanner;
6  public class lab_6_1 {
7      public static void main(String[] args){
8
9          System.out.println("Primul program in Java care citeste si afiseaza elementele unui vector");
10         int n;
11         Scanner input = new Scanner(System.in);
12         n = input.nextInt();
13         System.out.println("Numarul de elemente = "+n);
14         /* declararea variabilei de tip tablou unidimensional V[]*/
15
16         int V[] = new int[n];
17         // citirea elementelor in tabloul dat
18         int i;
19         for(i=0;i<n;i++)    V[i] = input.nextInt();
20
21         // afisarea elementelor tabloului
22         System.out.print("Vectorul citit este: ");
23         for(i=0;i<n;i++)    System.out.print(" "+V[i]);
24     }
25 }
26

```

Execute Mode, Version, Inputs & Arguments

JDK 17.0.1  Interactive Stdin Inputs

CommandLine Arguments

5  
1 2 3 4 5

Execute

Result

CPU Time: 0.23 sec(s), Memory: 37212 kilobyte(s)

```

Primul program in Java care citest si afiseaza elementele unui vector
Numarul de elemente = 5
Vectorul citit este: 1 2 3 4 5

```

2. Se citeste un numar n si un vector cu n componente numere intregi. Sa se afiseze valoarea maxima continuta in vector.

```

import java.util.Scanner;
public class lab_6_2 {
    public static void main(String[] args){
        int n;
        Scanner input = new Scanner(System.in);
        n = input.nextInt();
        System.out.println("Numarul de elemente = "+n);
        /* declararea variabilei de tip tablou unidimensional V[]*/

        int V[] = new int[n];
        // citirea elementelor in tabloul dat
        int i;
        for(i=0;i<n;i++)    V[i] = input.nextInt();

        int maxim=V[0];
        for(i=1;i<n;i++)
            if(maxim < V[i])    maxim = V[i];
    }
}

```

## Laborator - Programare orientată pe obiecte Limbajul Java

Adrian Runceanu

```
System.out.print("Valoarea maxima este = "+maxim);  
    }  
}
```

```
1  /*  
2  2. Se citeste un numar n si un vector cu n componente numere intregi.  
3  Sa se afiseze valoarea maxima continuta in vector.  
4  */  
5  import java.util.Scanner;  
6  public class lab_6_1 {  
7  public static void main(String[] args){  
8  
9      int n;  
10     Scanner input = new Scanner(System.in);  
11     n = input.nextInt();  
12     System.out.println("Numarul de elemente = "+n);  
13     /* declararea variabilei de tip tablou unidimensional V[]*/  
14  
15     int V[] = new int[n];  
16     // citirea elementelor in tabloul dat  
17     int i;  
18     for(i=0;i<n;i++)    V[i] = input.nextInt();  
19  
20     int maxim=V[0];  
21     for(i=1;i<n;i++)  
22         if(maxim < V[i])    maxim = V[i];  
23     System.out.print("Valoarea maxima este = "+maxim);  
24 }  
25 }  
26
```

### Execute Mode, Version, Inputs & Arguments

JDK 17.0.1

Interactive

Stdin Inputs

5  
1 2 13 4 5

CommandLine Arguments

 Execute



Result

CPU Time: 0.24 sec(s), Memory: 37656 kilobyte(s)

```
Numarul de elemente = 5  
Valoarea maxima este = 13
```

**Laborator - Programare orientată pe obiecte  
Limbajul Java****Adrian Runceanu**

3. Se citește un număr  $n$  și un vector cu  $n$  componente numere reale. Să se afișeze elementele vectorului sortate crescător. Se va utiliza **metoda sortării prin interschimbare**.

```
import java.util.Scanner;
public class lab_6_3 {
    public static void main(String[] args){
        int n;
        Scanner input = new Scanner(System.in);
        n = input.nextInt();
        System.out.println("Numarul de elemente = "+n);
        /* declararea variabilei de tip tablou unidimensional V[]*/

        double V[] = new double[n];
        // citirea elementelor in tabloul dat
        int i,j;
        double temp;
        for(i=0;i<n;i++) V[i] = input.nextInt();

        for(i=0;i<n-1;i++)
            for(j=i+1;j<n;j++)
                if(V[i] > V[j]){
                    temp = V[i];
                    V[i] = V[j];
                    V[j] = temp;
                }
        for(i=0;i<n;i++)
            System.out.print(V[i]+" ");
    }
}
```

## Laborator - Programare orientată pe obiecte Limbajul Java

Adrian Runceanu




```
1  /*
2  3. Se citeste un numar n si un vector cu n componente numere reale.
3  Sa se afiseze elementele vectorului sortate crescator.
4  Se va utiliza metoda sortarii prin interschimbare.
5  */
6  import java.util.Scanner;
7  public class lab_6_3 {
8      public static void main(String[] args){
9
10         int n;
11         Scanner input = new Scanner(System.in);
12         n = input.nextInt();
13         System.out.println("Numarul de elemente = "+n);
14         /* declararea variabilei de tip tablou unidimensional V[*]*/
15
16         double V[] = new double[n];
17         // citirea elementelor in tabloul dat
18         int i,j;
19         double temp;
20         for(i=0;i<n;i++)    V[i] = input.nextInt();
21
22
23         for(i=0;i<n-1;i++)
24             for(j=i+1;j<n;j++)
25                 if(V[i] > V[j])
26                     {
27                         temp = V[i];
28                         V[i] = V[j];
29                         V[j] = temp;
30                     }
31         for(i=0;i<n;i++)
32             System.out.print(V[i]+" ");
33     }
34 }
35
```

Execute Mode, Version, Inputs & Arguments

JDK 17.0.1  Interactive Stdin Inputs

CommandLine Arguments

7  
-90 123 456 -45 89 -1111 6789

**Execute**   

Result

CPU Time: 0.27 sec(s), Memory: 39292 kilobyte(s) compiled and e

```
Numarul de elemente = 7
-1111.0 -90.0 -45.0 89.0 123.0 456.0 6789.0
```

**Laborator - Programare orientată pe obiecte  
Limbajul Java****Adrian Runceanu**

4. Se citeste un numar n si un vector cu n componente numere intregi. Sa se calculeze si sa se afiseze suma elementelor din vector care sunt divizibile cu o valoare naturala a citita de la tastatura.

```
import java.util.Scanner;
public class lab_6_4 {
    public static void main(String[] args){
        int n, a;
        Scanner input = new Scanner(System.in);
        n = input.nextInt();
        System.out.println("Numarul de elemente = "+n);
        /* declararea variabilei de tip tablou unidimensional V[]*/

        int V[] = new int[n];
        // citirea elementelor in tabloul dat
        int i,j;
        for(i=0;i<n;i++) V[i] = input.nextInt();
        a = input.nextInt();
        System.out.println("Numarul a = "+a);

        int suma = 0;
        for(i=0;i<n;i++)
            if(V[i] % a == 0) suma = suma + V[i];

        System.out.print("Suma elementelor din tablou care sunt divizibile cu "+a+" este
        egala cu "+suma);
    }
}
```

## Laborator - Programare orientată pe obiecte Limbajul Java

Adrian Runceanu

```

1 import java.util.Scanner;
2 public class lab_6_4 {
3     public static void main(String[] args){
4         int n, a;
5         Scanner input = new Scanner(System.in);
6         n = input.nextInt();
7         System.out.println("Numarul de elemente = "+n);
8         /* declararea variabilei de tip tablou unidimensional V[]*/
9
10        int V[] = new int[n];
11        // citirea elementelor in tabloul dat
12        int i,j;
13        for(i=0;i<n;i++) V[i] = input.nextInt();
14        a = input.nextInt();
15        System.out.println("Numarul a = "+a);
16
17        int suma = 0;
18        for(i=0;i<n;i++)
19            if(V[i] % a == 0) suma = suma + V[i];
20
21        System.out.print("Suma elementelor din tablou care sunt divizibile cu "+a+" este egala cu "+suma);
22    }
23 }
24




```

Execute Mode, Version, Inputs & Arguments

JDK 17.0.1  Interactive Stdin Inputs

5  
12 4 75 90 33  
2

CommandLine Arguments

**Execute**   

Result

CPU Time: 0.23 sec(s), Memory: 37720 kilobyte(s) compiled and executed in 0.844

```

Numarul de elemente = 5
Numarul a = 2
Suma elementelor din tablou care sunt divizibile cu 2 este egala cu 106

```

5. Se citeste un numar n. Sa se calculeze si sa se afiseze un vector care sa contina cifrele numarului dat.

```

import java.util.Scanner;
public class lab_6_5 {
    private static Scanner input;
    public static void main(String[] args){
        int n;
        input = new Scanner(System.in);
        n = input.nextInt();
        System.out.println("Numarul de elemente = "+n);
        int V[] = new int[n]; /* declararea variabilei de tip tablou unidimensional V[]*/
        int i = 0;
        while(n != 0){
            V[i] = n % 10;
            n = n / 10;
            i++;}
        int cifre = i;
        for(i=0;i<cifre;i++)
            System.out.print(V[i]+" ");
    }
}

```

## Laborator - Programare orientată pe obiecte Limbajul Java

Adrian Runceanu

```
}  
}
```

```
1 import java.util.Scanner;  
2 public class lab_6_5 {  
3     private static Scanner input;  
4     public static void main(String[] args){  
5         int n;  
6         input = new Scanner(System.in);  
7         n = input.nextInt();  
8         System.out.println("Numarul de elemente = "+n);  
9         /* declararea variabilei de tip tablou unidimensional V[]*/  
10  
11         int V[] = new int[n];  
12  
13         int i = 0;  
14         while(n != 0)  
15         {  
16             V[i] = n % 10;  
17             n = n / 10;  
18             i++;  
19         }  
20         int cifre = i;  
21         for(i=0;i<cifre;i++)  
22             System.out.print(V[i]+" ");  
23     }  
24 }  
25 |
```

Execute Mode, Version, Inputs & Arguments

JDK 17.0.1  Interactive Stdin Inputs

CommandLine Arguments  67123

Result

CPU Time: 0.22 sec(s), Memory: 37384 kilobyte(s)

```
Numarul de elemente = 67123  
3 2 1 7 6
```

**Laborator - Programare orientată pe obiecte  
Limbajul Java****Adrian Runceanu**

6. Se citesc doua valori naturale n si m nenule, ce reprezinta numarul de elemente a doua multimi A si B. Sa se calculeze si apoi sa se afiseze multimea care reprezinta **reuniunea celor doua multimi**.

```
import java.util.Scanner;
public class lab_6_6 {
    private static Scanner input;
    public static void main(String[] args){
        int n,m,i,j,k;
        input = new Scanner(System.in);
        n = input.nextInt();
        System.out.println("Numarul de elemente ale multimii A = "+n);
        /* declararea variabilei de tip tablou unidimensional A[]*/
        int A[] = new int[n];
        // citirea elementelor in tabloul dat
        for(i=0;i<n;i++) A[i] = input.nextInt();

        m = input.nextInt();
        System.out.println("Numarul de elemente ale multimii B = "+m);
        /* declararea variabilei de tip tablou unidimensional B[]*/
        int B[] = new int[m];
        // citirea elementelor in tabloul dat
        for(i=0;i<m;i++) B[i] = input.nextInt();

        /* declararea variabilei de tip tablou unidimensional C[]*/
        int C[] = new int[200];

        // copiem elementele din multimea A in multimea C
        for(i=0;i<n;i++) C[i] = A[i];

        k = n; // k reprezinta numarul de elemente din multimea C
        for(j=0; j<m; j++){
            boolean ok = true;
            for(i=0;i<n;i++)
                if(B[j] == A[i]) ok = false;
            if(ok == true){
                C[k] = B[j];
                k++;
            }
        }
    }
}
```

## Laborator - Programare orientată pe obiecte Limbajul Java

Adrian Runceanu

```

    for(i=0;i<k;i++)
        System.out.print(C[i]+" ");
    }
}

```

```

1  import java.util.Scanner;
2  public class lab_6_6 {
3      private static Scanner input;
4  public static void main(String[] args){
5      int n,m,i,j,k;
6      input = new Scanner(System.in);
7      n = input.nextInt();
8      System.out.println("Numarul de elemente ale multimii A = "+n);
9      /* declararea variabilei de tip tablou unidimensional A[*/
10     int A[] = new int[n];
11     // citirea elementelor in tabloul dat
12     for(i=0;i<n;i++)    A[i] = input.nextInt();
13
14     m = input.nextInt();
15     System.out.println("Numarul de elemente ale multimii B = "+m);
16     /* declararea variabilei de tip tablou unidimensional B[*/
17     int B[] = new int[m];
18     // citirea elementelor in tabloul dat
19     for(i=0;i<m;i++)    B[i] = input.nextInt();
20
21     /* declararea variabilei de tip tablou unidimensional C[*/
22     int C[] = new int[200];
23
24     // copiem elementele din multimea A in multimea C
25     for(i=0;i<n;i++) C[i] = A[i];
26
27     k = n; // k reprezinta numarul de elemente din multimea C
28     for(j=0; j<m; j++){
29         boolean ok = true;
30         for(i=0;i<n;i++)
31             if(B[j] == A[i])    ok = false;
32         if(ok == true){
33             C[k] = B[j];
34             k++;
35         }
36     }
37     for(i=0;i<k;i++)
38         System.out.print(C[i]+" ");
39 }
40 }
41

```

## Laborator - Programare orientată pe obiecte Limbajul Java

Adrian Runceanu

Execute Mode, Version, Inputs & Arguments

JDK 17.0.1  Interactive

Stdin Inputs

3  
1 3 5  
5  
2 3 4 5 6

CommandLine Arguments

Execute

Result

CPU Time: 0.24 sec(s), Memory: 37540 kilobyte(s)

```

Numarul de elemente ale multimii A = 3
Numarul de elemente ale multimii B = 5
1 3 5 2 4 6

```

### Probleme propuse spre rezolvare

**Lab6\_1:** Se citeste un numar  $n$  si un vector cu  $n$  componente numere intregi. Sa se afiseze valoarea minima continuta in vector.

**Lab6\_2:** Se citesc doua valori naturale  $n$  si  $m$  nenule, ce reprezinta numarul de elemente a doua multimi  $A$  si  $B$ . Sa se calculeze si apoi sa se afiseze multimea care reprezinta intersectia celor doua multimi.

**Lab6\_3:** Se citesc doua valori naturale  $n$  si  $m$  nenule, ce reprezinta numarul de elemente a doua multimi  $A$  si  $B$ . Sa se calculeze si apoi sa se afiseze multimea care reprezinta diferenta  $A-B$  a celor doua multimi.

**Lab6\_4:** Se citeste un numar  $n$  si un vector cu  $n$  componente numere intregi. Sa se verifice daca elementele vectorului sunt sau nu in ordine crescatoare.

**Lab6\_5:** Se considera un vector  $\mathbf{x} = \{ x_1, x_2, x_3, \dots, x_n \}$ . Sa se scrie un program Java care sa modifice vectorul astfel incat in final sa avem:

a)  $\mathbf{x} = \{ x_2, x_3, \dots, x_n, x_1 \}$ ; b)  $\mathbf{x} = \{ x_n, x_1, x_2, x_3, \dots, x_{n-1} \}$ ; c)  $\mathbf{x} = \{ x_2, x_1, x_4, x_3, \dots, x_n, x_{n-1} \}$

**Lab6\_6:** Se considera doi vectori  $\mathbf{x}$  si  $\mathbf{y}$  cu cate  $n$  elemente numre intregi fiecare. Sa se scrie un program Java care sa calculeze urmatoarele valori:

a)  $e = (x_1+y_1) \times (x_2+y_2) \times \dots \times (x_n+y_n)$ ; b)  $e = x_1y_1 + x_2y_2 + \dots + x_ny_n$ ;

## Laborator - Programare orientată pe obiecte Limbajul Java

Adrian Runceanu

$$c) e = x_1^2 y_1 + x_2^2 y_2 + \dots + x_n^2 y_n$$

**Lab6\_7:** Se considera un vector  $\mathbf{x}$  care are elemente numere reale. Sa se scrie un program Java care sa introduca media geometrica intre oricare doua elemente vecine din vector. Vectorul se va afisa inainte si dupa modificare.

**Lab6\_8:** Se considera un vector  $\mathbf{x}$  cu  $n$  numere intregi. Sa se copieze intr-un alt vector  $\mathbf{y}$  elementele pozitive ale vectorului initial.

**Lab6\_9:** Se citeste un tablou unidimensional cu  $n$  ( $1 \leq n \leq 100$ ) componente numere naturale. Se cere sa se construiasca si sa se afiseze un nou vector cu componentele *patrate perfecte* ale vectorului initial.

**Lab6\_10:** Sa se rearanjeze elementele unui tablou unidimensional de numere intregi, astfel incat numerele pare sa apara inaintea numerelor impare. In cadrul subsecventei de numere pare, respectiv impare, elementele trebuie sa apara in ordinea in care erau in vectorul initial.

**Lab6\_11:** Se considera un tablou unidimensional  $x$  cu  $n$  numere intregi. Se cere sa se calculeze cel mai mare divizor comun al lor.

**Lab6\_12:** Se da un tablou unidimensional  $X$  cu  $n$  numere reale si se cere sa se modifice astfel incat sa se intercaleze intre oricare doua elemente consecutive, media lor aritmetica.

## 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] Tudor Sorin, Vlad Hutanu, **Bazele programarii in Java**, Editura L&S Info-Mat, Bucuresti, 2005.

[4] Doina Logofatu, **Algoritmi fundamentali in Java. Aplicatii** – Editura Polirom, Iasi, 2007.

[5] Horia Georgescu, **Introducere in universul Java**, Editura Tehnica, Bucuresti, 2002.

[6] Adrian Runceanu, Gheorghe Gîlcă, Alina Cotoi, **Aplicații JAVA - introducere teoretică și practică**, Editura Academica Brâncuși din Târgu-Jiu, 2017.