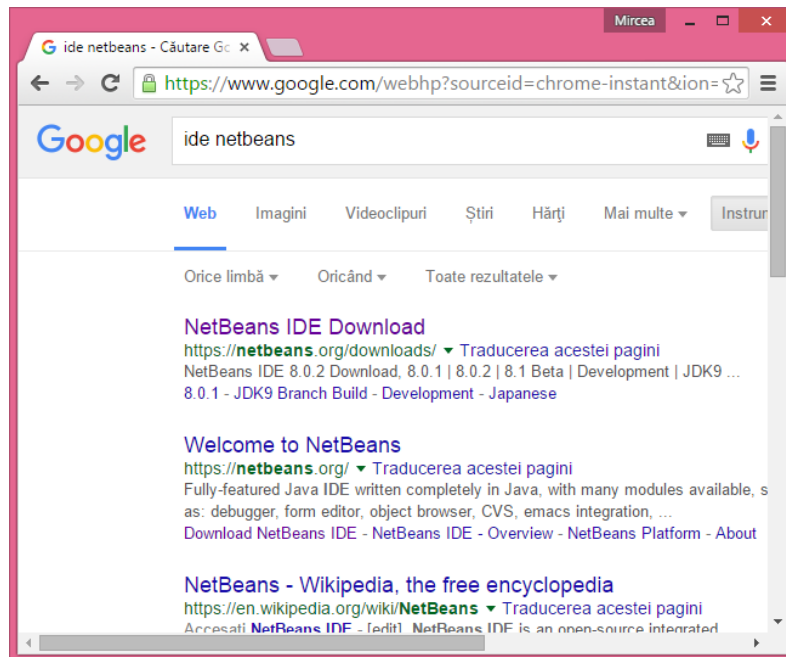


LUCRARE PRACTICĂ NR. 2

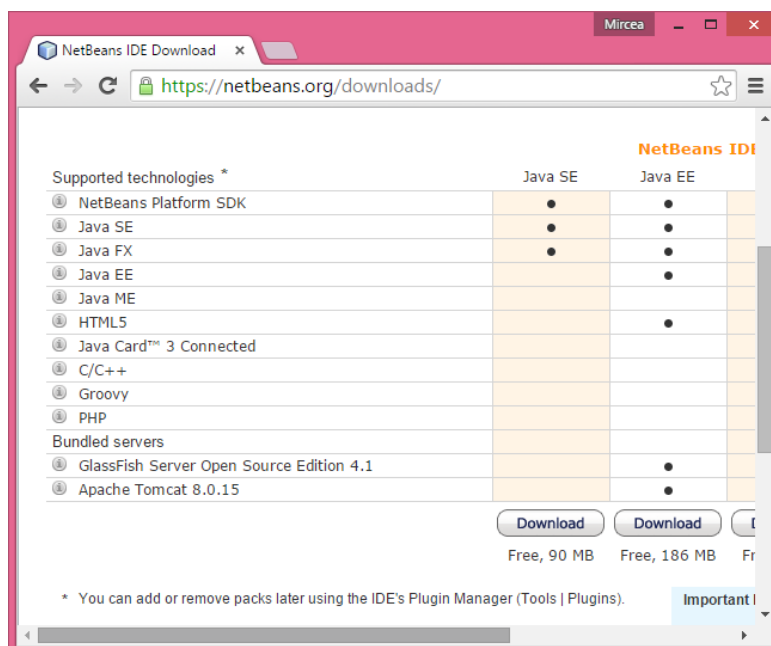
Programarea Java

1. Instalarea IDE Netbeans

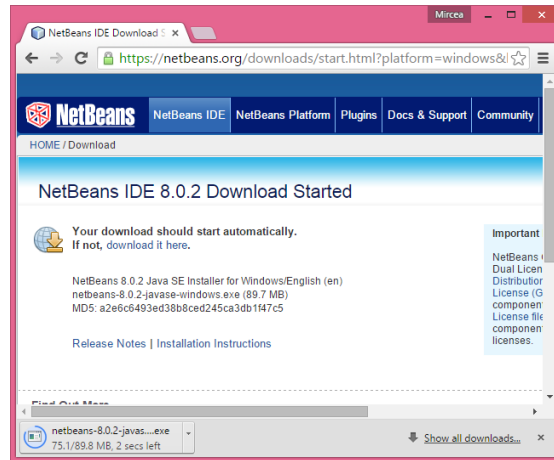
a. Lansarea browserului și scrierea interogării



b. Trecerea pe prima referință apărută



- c. Apăsăm butonul sub colonița Java SE. Se va începe descărcarea IDE Netbeans împreună cu JDK.



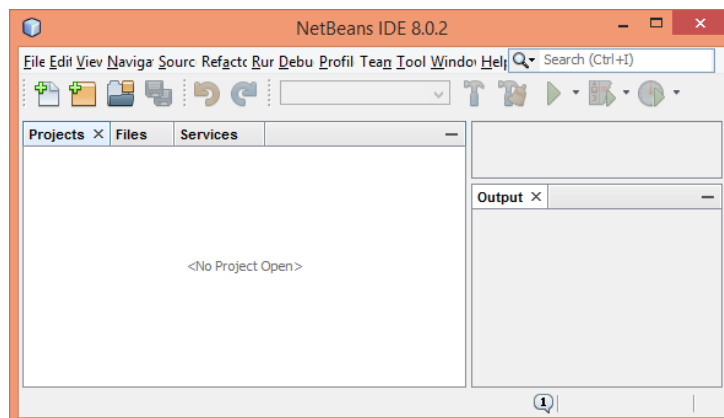
- f. După ce s-a descărcat, se va lansa instalarea IDE Netbeans. Vor apărea ferestre de dialog pentru a instala.

2. Exemplu de elaborare a primului program în Java cu mijloace IDE NetBeans

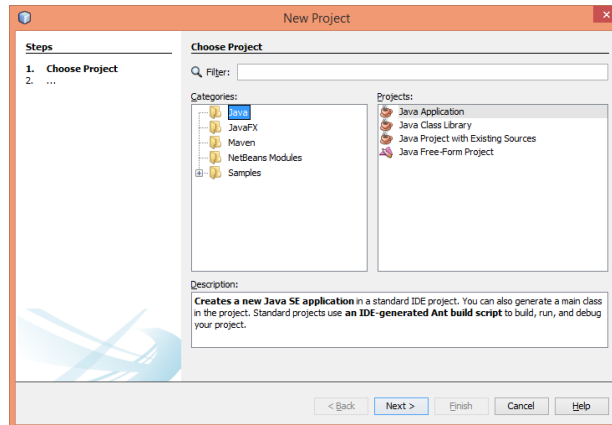
- a. Pentru a lansa IDE NetBeans se face double click pe shortcut-ul de pe masa de lucru. În rezultat se va începe pornirea IDE NetBeans.



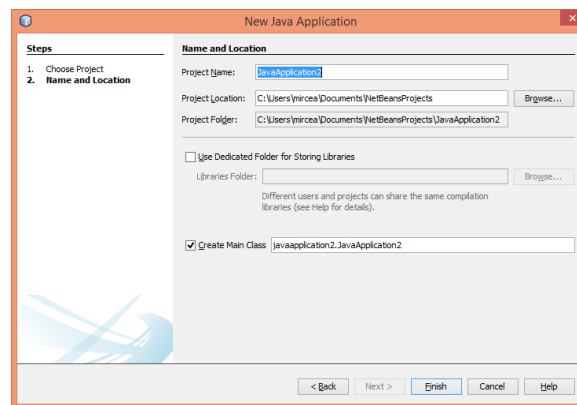
- b. După imaginea de mai sus, dacă porniți prima dată IDE NetBeans va apărea o fereastră de dialog, în care trebuie de indicat mapa unde se vor păstra proiectele create în IDE NetBeans.



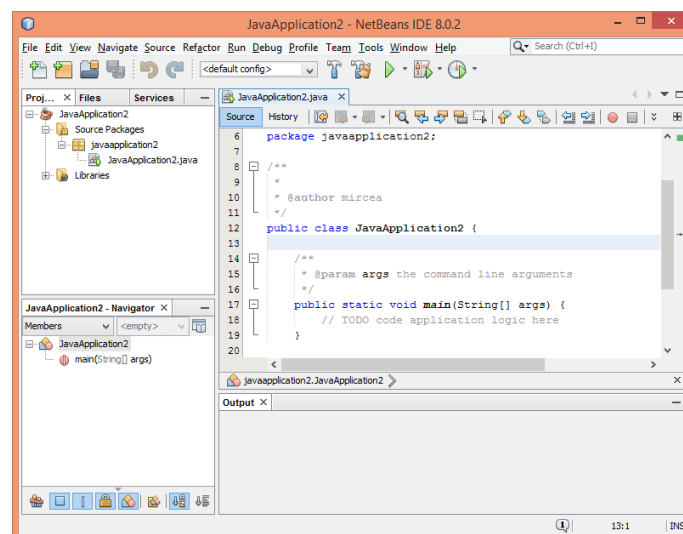
- c. Acum sîntem gata pentru prima aplicație Java în IDE NetBeans. În meniul File, opțiunea New Project.



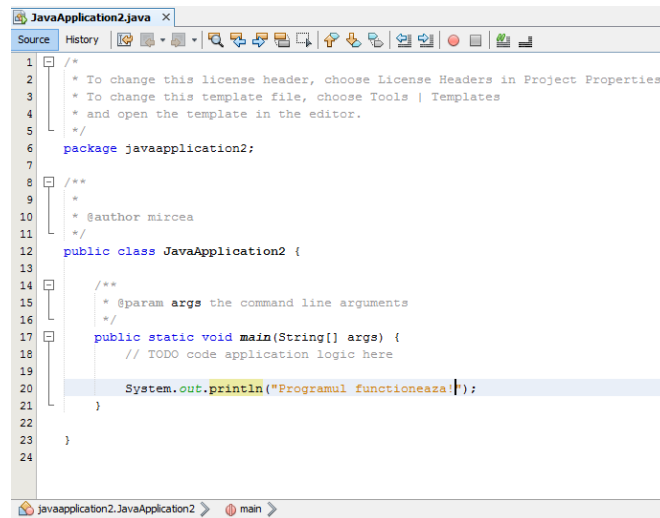
- d. Selectăm în fila Categories – Java, și în Project - Java Application. Apoi apăsăm butonul Next >.



- e. După ce a fost setat numele proiectului, se poate de apăsă pe butonul Finish.

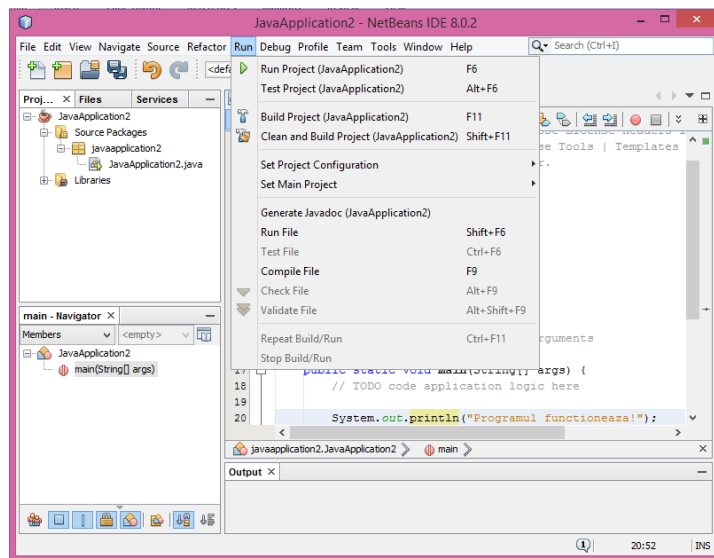


f. Observăm proiectul conține deja fișierul în care va fi scris programul Java, încadrat într-un pachet. Rămîne doar de scris codul programului. În final codul va arăta astfel:

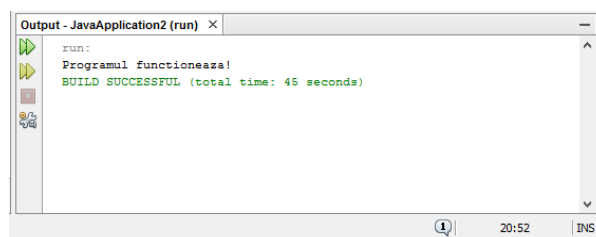


```
1  /*
2  * To change this license header, choose License Headers in Project Properties.
3  * To change this template file, choose Tools | Templates
4  * and open the template in the editor.
5  */
6  package javaapplication2;
7
8  /**
9   *
10  * @author mircea
11  */
12  public class JavaApplication2 {
13
14      /**
15       * @param args the command line arguments
16       */
17      public static void main(String[] args) {
18          // TODO code application logic here
19
20          System.out.println("Programul functioneaza!");
21      }
22  }
23
24
```

h. Pentru a lansa programul se apasă pe butonul corespunzător comenzii Run din bara de instrumente, sau din meniul Run se alege comanda Run file sau Run Project.

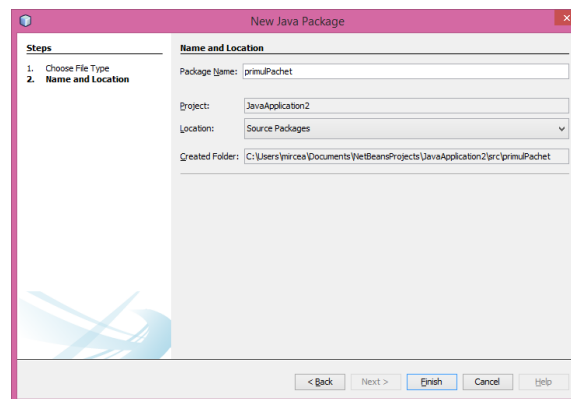
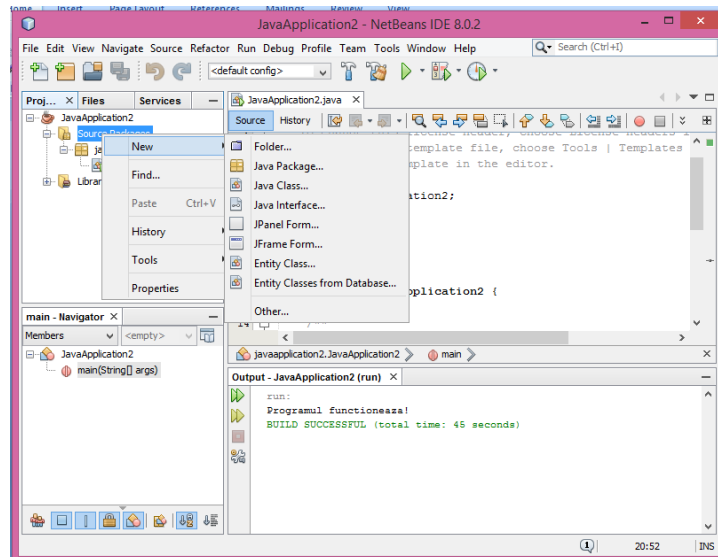


i. În final se va observa în zona de jos Output efectul execuției programului.

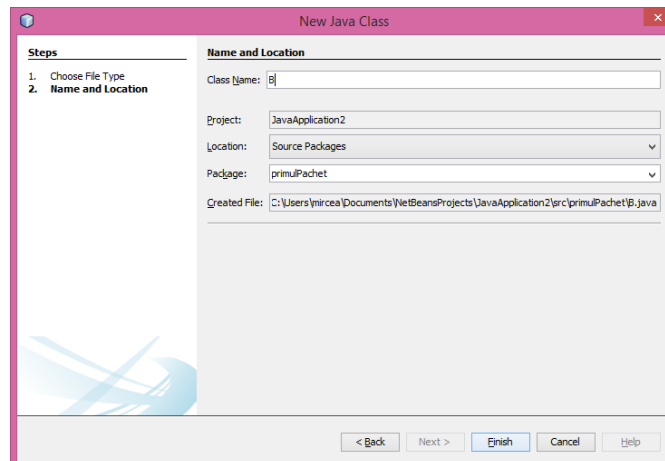
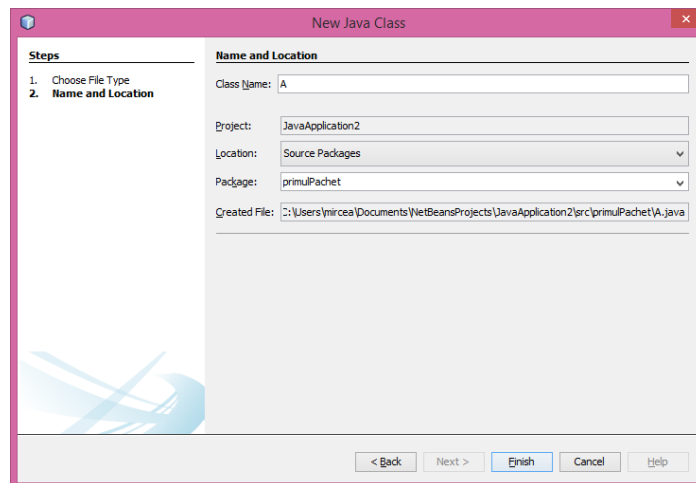
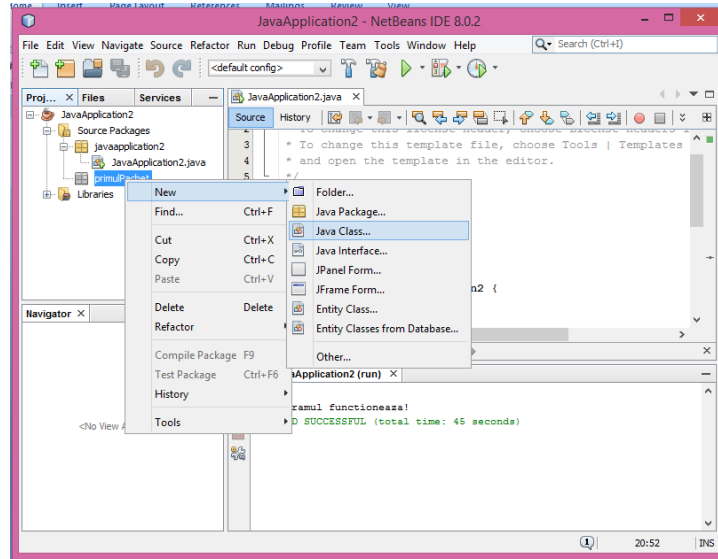


3. Exemplu de elaborare a pachetului cu două clase A și B.

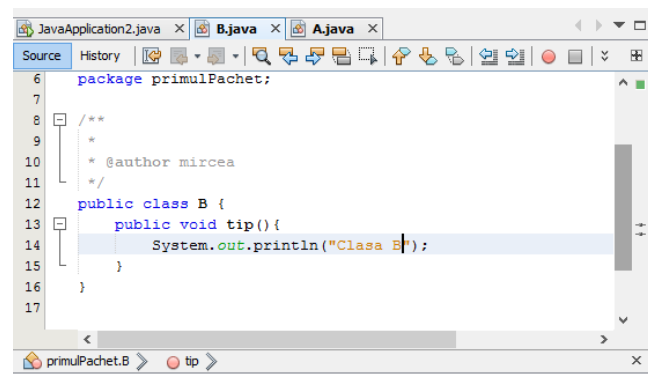
- a. În continuare vom crea un nou pachet în proiectul JavaApplication2. Pentru aceasta fiind selectată mapa Source Package, se face un click cu butonul drept pe această mapă, se alege opțiunea New → Java Package. În fereastra care a apărut se indică numele pachetului (primulPachet – în cazul nostru).



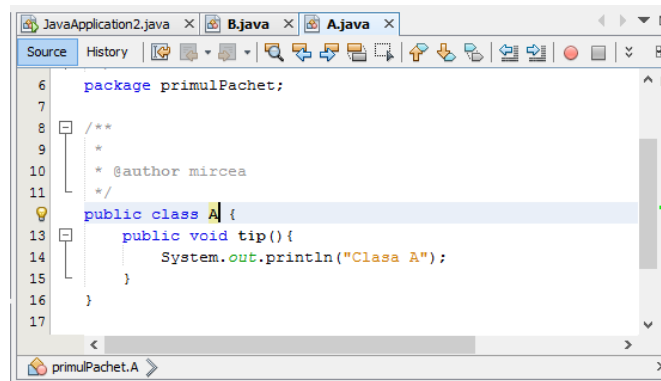
- b. În acest pachet creăm două clase A și B. Se selectează pachetul, se face un click dreapta pe el. În meniul apărut se selectează opțiunea New → Java Class. În fereastra de dialog apărută se indică numele clasei (în cazul nostru A și B).



c. Creăm cele două fișiere sursă A.java și B.java.

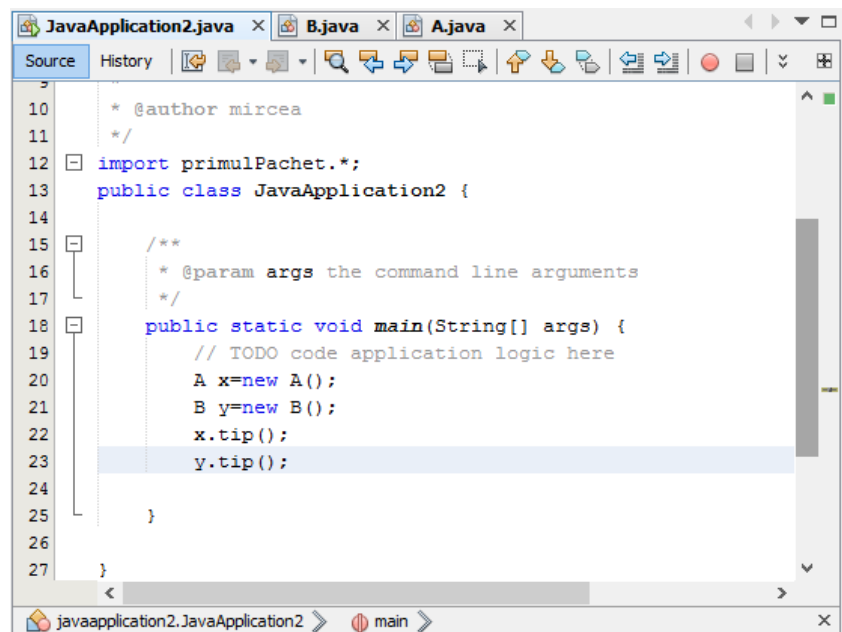


```
6 package primulPachet;
7
8 /**
9  *
10  * @author mircea
11  */
12 public class B {
13     public void tip() {
14         System.out.println("Clasa B!");
15     }
16 }
17
```



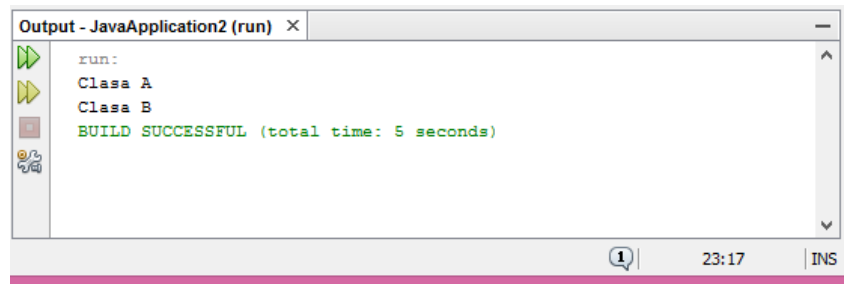
```
6 package primulPachet;
7
8 /**
9  *
10  * @author mircea
11  */
12 public class A {
13     public void tip() {
14         System.out.println("Clasa A!");
15     }
16 }
17
```

d. Revenim în clasa principală și culegem textul pentru a lucra cu clasele A și B din pachetul primulPachet. În acest caz metoda main și clasa prima vor arăta după cum urmează.



```
10  * @author mircea
11  */
12 import primulPachet.*;
13 public class JavaApplication2 {
14
15     /**
16      * @param args the command line arguments
17      */
18     public static void main(String[] args) {
19         // TODO code application logic here
20         A x=new A();
21         B y=new B();
22         x.tip();
23         y.tip();
24     }
25 }
26
27
```

- e. După ce lansăm în execuție programul vom observa mesajele corespunzătoare metodelor `tip()` din fiecare program.



4. Sarcini pentru antrenament

- Găsiți cum se poate de elaborat un subpachet în IDE NetBeans. Creați în exemplul de mai sus un subpachet în pachetul `primulPachet`, cu clasa `C`, care are o metodă `tip` pentru afișarea mesajului clasa `C`. Completați metoda `main` al clasei `prima` cu instrucțiunile corespunzătoare apelului metodei `tip`.
 - **ELABORAȚI UN PACHET CU NUMELE LABORATOR** care să conțină câte o clasă în care va fi o metodă care rezolvă câte o problemă din cele de mai jos.
 - **APOI ELABORAȚI UN PROGRAM CARE APELEAZĂ clasele din pachetul LABORATOR.**
- a. Se citesc n numere reale. Să se afișeze valoarea minimă citită.
 - b. Se citește un șir de numere întregi până la întâlnirea numărului 0 . Să se calculeze media aritmetică a numerelor din șir.
 - c. Se citește un număr natural cu 5 cifre. Afișați numpărul format după eliminarea cifrei din mijloc.
 - d. Se citește un vector cu n componente, numere naturale. Să se afișeze cel mai mare număr rațional subunitar în care numărătorul și numitorul fac parte din mulțimea valorilor citite. Exemplu: dacă am citit valorile 1 2 3 se afișează $2/3$.

- e. Se citește un vector cu n componente numere naturale. Se cere să se obțină toate permutările circulare la dreapta.

Exemplu: dacă $n=4$ și vectorul este 1 2 3 4, permutările circulare sînt: 1 2 3 4, 4 1 2 3, 3 4 1 2, 2 3 4 1.