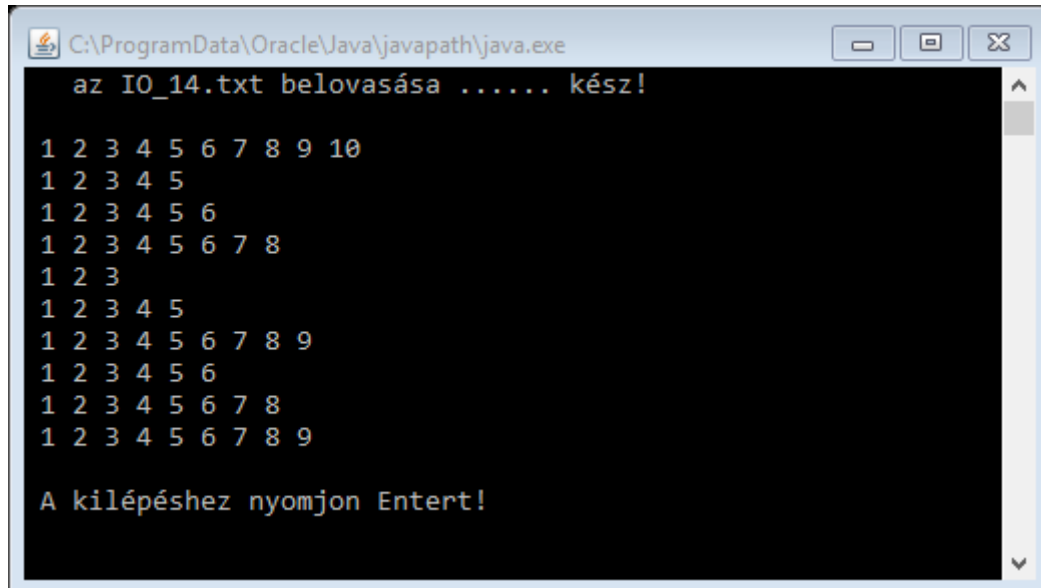


Feladat

Írj programot, mely beolvassa a 10 elemet tartalmazó **IO_14.txt** állomány vesszővel elválasztott adatait, melyek pozitív egész számok, és kiírja 1-től N-ig [1, N] az egész számokat a képernyőre *-1-től eddig a számig-*, egymástól szóközzel elválasztva a minta szerint.



```
az IO_14.txt beolvasása ..... kész!

1 2 3 4 5 6 7 8 9 10
1 2 3 4 5
1 2 3 4 5 6
1 2 3 4 5 6 7 8
1 2 3
1 2 3 4 5
1 2 3 4 5 6 7 8 9
1 2 3 4 5 6
1 2 3 4 5 6 7 8
1 2 3 4 5 6 7 8 9

A kilépéshez nyomjon Entert!
```

Durva algoritmus

globális t[10] // tagváltozó

f1():

segéd változóba beolvasni

split()-el t[]-be [segítség](#) ha kell

f2():

N = t.hossza;

külső ciklus i=0-tól egyesével N-ig

 N1=t[i];

 belső ciklus j=1-től egyesével (N1+1)-ig

 Ki: j

 belső ciklus vége

külső ciklus vége

h1

```
1 package io_14;
2 // split, Integer.parseInt gyakorlása 1.
3 import java.io.*; // IO stream
4 public class IO_14 {
5     static String t[]=new String[10]; // globális 10 hosszú tömb
6     static void f1(){
7         RandomAccessFile f; String sor;
8         try{//megpróbálom
9             f = new RandomAccessFile("IO_14.txt","r");
10            sor=f.readLine();
11            t=sor.split(",");// a t tömb feltöltése a split-tel
12            f.close();
13        }catch(IOException e){//a hiba elkapása
14            System.out.println("Baj van babám");
15        }
16        System.out.println("    "+"az IO_14.txt belovasása ..... "+" kész!");
17        System.out.println();
18    } // f1 vége
```

[Vissza](#)