

ANS Elbląg

**Instytut Informatyki Stosowanej im. Krzysztofa
Brzeskiego**

Programowanie obiektowe I – laboratorium

Studium Stacjonarne, sem. 3, 2022/2023

Sprawozdanie nr : 3,

nr grupy: 1,

dzień: wtorek,

godz. 12:00.

Data wykonania ćwiczenia:28.10

Data oddania sprawozdania: 13.11

Nazwisko i imię: Kuczawski Kacper

Nr albumu: 20195

Nazwa pliku : lab3_kuczawski_kacper20195

1. W klasie publicznej **Lab3Z1** zdefiniować tablicę jednowymiarową **tab** liczb typu **double** oraz metodę statyczną **readTab1D** wczytującą tablicę liczb typu **double** oraz metodę statyczną **printTab1D** drukującą tablicę liczb typu **double**, w metodzie **main()** wczytać i wydrukować tablicę **x[5]** przy zastosowaniu tych metod.

Uwaga. Wywołanie metod statycznych klasy nie wymaga tworzenia obiektu klasy **Lab3Z1**, metody są definiowane z użyciem słowa **static** i wywoływanie jako **Lab3Z1.readTab1D(parametry)**.

Oraz

2. Do klasy **Lab3Z1** dodać metodę statyczną **maxMin** wyznaczającą wartości minimalną i maksymalną w tablicy jednowymiarowej typu **double** i zwracającą te wartości przy użyciu dodatkowej tablicy jednowymiarowej. Wywołać metodę w metodzie **main()** dla tablicy **x[5]** (z zadania 1).

Treść:

```
import java.io.*;
import java.util.*;
import java.text.*;
public class Lab3Z1{
    public static void main(String args[]){
        double x[] = new double[5];
        int kol;
        kol=x.length;
        Lab3Z1.readTab1D(kol,x);
        Lab3Z1.printTab1D(x);
        Lab3Z1.maxMin(x);
    }
    public static void readTab1D(int n, double tab[]){
        Scanner sc=new Scanner(System.in);
        for(int i=0; i<n; i++){
            System.out.print("Podaj " + (i+1) + " wartosc:");
            tab[i]=sc.nextDouble();
        }
    }
    public static void printTab1D(double tab[]){
        for(double element:tab) System.out.println(element);
    }
    public static void maxMin(double tab[]){
        int kol;
        kol=tab.length;
        double temp, min, max;
        min=tab[0];
        max=tab[0];
        double wyniki[]=new double[2];
        for(int i=0; i<kol; i++){
            temp=tab[i];
            if(temp>max){
                max=temp;
            } else if(temp<min){
                min=temp;
            }
        }
    }
}
```

```

        }
        wyniki[0]=min;
        wyniki[1]=max;
        System.out.print("Minimalna wartosc tablicy: " + wyniki[0] + " "
maksymalna:" + wyniki[1]);
    }
}

```

Wyniki:

```

Podaj 1 wartosc:3
Podaj 2 wartosc:5
Podaj 3 wartosc:6
Podaj 4 wartosc:9
Podaj 5 wartosc:2
3.0
5.0
6.0
9.0
2.0
Minimalna wartosc tablicy: 2.0 maksymalna:9.0
Process finished with exit code 0

```

3. Zdefiniować klasę publiczną **Lab3Z3** oraz w tym samym pliku klasę **Tab2DReadPrint** zawierającą metody **readTab2D**, **printTab2D** i **transpose2D**. Metody te odpowiednio wczytują, drukują i transponują tablicę dwuwymiarową typu **int**. W metodzie **main()** klasy **Lab3Z3** stworzyć obiekt klasy **Tab2DReadPrint** oraz wczytać i wydrukować tablicę dwuwymiarową liczb **y[3][3]** typu **int**, następnie dokonać jej transpozycji i ponownie wydrukować.

Treść:

```

import java.io.*;
import java.util.*;
import java.text.*;
public class Lab3Z3{
    public static void main(String args[]){
        int y[][]=new int[3][3];
        int kol, wier;
        kol=y.length;
        wier=y[0].length;
        Tab2DReadPrint.readTab2D(kol, wier, y);
        Tab2DReadPrint.printTab2D(y);
    }
}
class Tab2DReadPrint{
    public static void readTab2D(int w, int k, int tab[][]){
        Scanner sc=new Scanner(System.in);
        for(int i=0; i<w; i++){

```

```

        for(int j=0; j<k; j++){
            System.out.print("Podaj tab[" + (i+1) + "][" + (j+1) + "]: ");
            tab[i][j]=sc.nextInt();
        }
    }
}

public static void printTab2D(int tab[][][]){
    for(int x[]:tab){
        for(int y: x){
            System.out.println(y+" ");
        }
        System.out.println();
    }
}
}

```

4. Utworzyć klasę publiczną **Lab3Z4** i klasę **Pierwsza** z metodą prywatną **silnia** oraz metodą **Oblicz** obliczającą wartość sumy odwrotności silni kolejnych n liczb całkowitych. W klasie **Lab3Z4** utworzyć obiekt klasy **Pierwsza** i wywołać dla niego metodę **Oblicz**, Wydrukować wynik.

Uwaga. Metoda prywatna może być wywołana tylko w ramach klasy, do której należy.

W klasie **Pierwsza** w metodzie **Oblicz** należy wywołać odpowiednio metodę **silnia** i przekazać z metody otrzymaną wartość sumy odpowiednich odwrotności silni.

Treść:

```

import java.io.*;
import java.util.*;
import java.text.*;
public class temp{
    public static void main(String args[]){
        int x;
        System.out.print("\nPodaj x:");
        Scanner sc=new Scanner(System.in);
        x=sc.nextInt();
        //int sil= new Pierwsza.silnia(x);
        //double sum= new Pierwsza.Oblicz(x);
        double sum;
        sum=Pierwsza.Oblicz(x);
        System.out.print("\nSuma odwrotnosci silni:"+sum);
    }
}
class Pierwsza{
    private static int silnia(int n){
        int silnia=1;
        for(int i=1;i<(n+1);i++){
            silnia=silnia*i;
        }
        //System.out.print("\nSilnia="+silnia);
        return silnia;
    }
}

```

```
public static double Oblicz(int n){  
    double suma=0;  
    double mian;  
    for(int i=1;i<(n+1);i++){  
        mian=Pierwsza.silnia(i);  
        suma=suma+(1/mian);  
    }  
    System.out.print("\nSuma odwrotnosci silni:"+suma);  
    return suma;  
}  
}
```

Wyniki:

```
Podaj x:5  
  
Suma odwrotnosci silni:1.7166666666666668  
Silnia=120  
Suma odwrotnosci silni:1.7166666666666668  
Process finished with exit code 0
```