

ANS Elbląg

**Instytut Informatyki Stosowanej im. Krzysztofa
Brzeskiego**

Programowanie obiektowe I – laboratorium

Studium Stacjonarne, sem. 3, 2022/2023

Sprawozdanie nr : 8,

nr grupy: 1,

dzień: wtorek,

godz. 12:00.

Data wykonania ćwiczenia: 13.12

Data oddania sprawozdania: 30.01

Nazwisko i imię: Kuczawski Kacper

Nr albumu: 20195

Nazwa pliku : lab8_kuczawski_kacper20195

1. Napisać klasę **Point** z dwoma polami **x,y** (współrzędne punktu na płaszczyźnie) typu **int**, dwoma konstruktorami (konstruktorem bezparametrowym i konstruktorem z dwoma parametrami) oraz metodami: **wyzeruj**(zeruje pola obiektu), **przesun**(zmienia wartości współrzędnych) i **wypisz**. W metodzie **main()** klasy publicznej **Lab8Zad1** stworzyć obiekty **punkt1** i **punkt2** klasy **Point** (**punkt1** – konstruktor bezparametryw i **punkt2** – konstruktor z parametrami (10,20)). Obiektowi **punkt2** nadać wartości współrzędnych (5,15). Wydrukować współrzędne obu obiektów. Sprawdzić metodę **wyzeruj** dla obiektu i wydrukować wartości jego pól po jej wywołaniu.

Treść:

```
import java.io.*;
import java.util.*;
import java.text.*;
public class temp{
    public static void main(String args[]){
        Point p = new Point(1,2);
        p.wypisz();
        p.wyzeruj();
        p.wypisz();
        p.przesun(3,4);
        p.wypisz();
        Point pb = new Point();
        pb.wypisz();
        pb.wyzeruj();
        pb.wypisz();
        pb.przesun(1,2);
        pb.wypisz();
    }
}
class Point{
    int x;
    int y;
    Point(){
        this(3,4);
    }
    Point(int a, int b){
        x=a;
        y=b;
    }
    void wyzeruj(){
        x=0;
        y=0;
    }
    void przesun(int a, int b){
        x=a;
        y=b;
    }
    void wypisz(){
        System.out.print("\nPunkt o współrzędnych x: "+x+" y: "+y+"\n");
    }
}
```

```
    }  
}
```

Wyniki:

```
Punkt o współrzędnych x: 1 y: 2
```

```
Punkt o współrzędnych x: 0 y: 0
```

```
Punkt o współrzędnych x: 3 y: 4
```

```
Punkt o współrzędnych x: 3 y: 4
```

```
Punkt o współrzędnych x: 0 y: 0
```

```
Punkt o współrzędnych x: 1 y: 2
```

```
Process finished with exit code 0
```

2. Napisać klasę **Box** z polami **długość**, **szerokość**, **wysokość** z konstruktorem o trzech parametrach oraz metodą **objętość** obliczającą objętość pudełka oraz metodę **sześciian** obliczającą bok sześciianu o zadanej objętości. W klasie publicznej **Lab8Zad2** stworzyć przy użyciu konstruktora obiekt **pudełko** klasy **Box** i obliczyć jego objętość, następnie utworzyć pudełko sześciennie o zadanej objętości. Wydrukować wyniki.

Treść:

```
import java.io.*;  
import java.util.*;  
import java.text.*;  
import java.math.*;  
public class temp{  
    public static void main(String args[]){  
        Box p = new Box(3.5,10.0,5.0);  
        double V, a;  
        p.wypisz();  
        V = p.V();  
        a = p.cube(27);  
        System.out.print("\nObjętość prostopadłościanu: "+V);  
        System.out.print("\nDługość boku sześciianu o V=27: "+a);  
    }  
}  
class Box{  
    double a;  
    double b;
```

```

double c;
Box(double x, double y, double z){
    a=x;
    b=y;
    c=z;
}
double v(){
    double v;
    v=a*b*c;
    return v;
}
double cube(double v){
    double x;
    x=Math.cbrt(v);
    return x;
}
void wypisz(){
    System.out.print("\nProstopadłościan o wymiarach a: "+a+" b: "+b+" c: "
"+c+"\n");
}
}

```

Wyniki:

```

Prostopadłościan o wymiarach a: 3.5 b: 10.0 c: 5.0

Objętość prostopadłościanu: 175.0
Długość boku sześcianu o V=27: 3.0
Process finished with exit code 0

```

3. Napisać klasę **Student1** z polami **nazwisko**, **imie**, **ocena1**, **ocena2**, **srednia** oraz metodami **srednia**(oblicza średnią arytmetyczną ocen), **wypisz** (wypisuje nazwisko, imię, obie oceny i średnią tych ocen). W klasie publicznej **Uczelnia1** stworzyć obiekt klasy **Student1**, przypisać mu oceny (wczytać z wejścia poszczególne pola (pola **ocena1**, **ocena2** z zabezpieczeniem formatu)), wypisać dane studenta, oceny oraz średnią arytmetyczną.

Treść:

```

import java.io.*;
import java.util.*;
import java.text.*;
import java.math.*;
import java.text.*;
public class temp{
    public static void main(String args[]){
        Scanner sc=new Scanner(System.in);
        System.out.print("Podaj ocenę 1:");
        while(! sc.hasNextDouble()){
            System.out.print("Blad, podaj poprawnie:");
            sc.next();
        }
    }
}

```

```

        double o1=sc.nextDouble();
        System.out.print("Podaj ocenę 2:");
        while(! sc.hasNextDouble()){
            System.out.print("Blad, podaj poprawnie:");
            sc.next();
        }
        double o2=sc.nextDouble();
        Student S = new Student("Faworny", "Franciszek",o1,o2);
        S.wypisz();
    }
}

class Student{
    String nazwisko;
    String imie;
    double o1;
    double o2;
    double srednia;
    Student(String n, String i, double x, double y){
        nazwisko = n;
        imie     = i;
        o1       = x;
        o2       = y;
        srednia = srednia();
    }
    double srednia(){
        double S;
        S=(o1+o2)/2;
        return S;
    }
    void wypisz(){
        System.out.print("\nNazwisko: "+nazwisko);
        System.out.print("\nImię: "+imie);
        System.out.print("\nOcena 1: "+o1);
        System.out.print("\nOcena 2: "+o2);
        System.out.print("\nŚrednia: "+srednia);
    }
}

```

Wyniki:

```

Podaj ocenę 1:4
Podaj ocenę 2:5

Nazwisko: Faworny
Imię: Franciszek
Ocena 1: 4.0
Ocena 2: 5.0
Średnia: 4.5
Process finished with exit code 0

```

4. Napisać klasę **Student2** (zmodyfikowana wersja klasy **Student1**) z polami **ocena1**, **ocena2** zmienionymi na pola prywatne i dopisać metodę **wstawOcene**, która będzie wstawiać każdą z ocen i nie pozwoli, żeby została wpisana wartość większa od 5 lub mniejsza od 2. Metoda powinna zwracać wartość typu **boolean** i powinna powodować kolejną próbę wprowadzenia oceny, jeśli zwrócona wartość jest równa **false**. W klasie publicznej **Uczelnia2** utworzyć obiekt klasy **Student2** (konstruktor z parametrami **nazwisko** i **imie**), i wywołać dla niego metody **wstawOcenę** i **średnia**. Wydrukować wyniki.

Treść:

```
import java.io.*;
import java.util.*;
import java.text.*;
import java.math.*;
import java.text.*;
public class temp{
    public static void main(String args[]){
        Scanner sc=new Scanner(System.in);
        Student S = new Student("Faworny", "Franciszek");
        double o1, o2;
        do{
            System.out.print("Podaj ocenę 1:");
            while(! sc.hasNextDouble()){
                System.out.print("Blad, podaj poprawnie:");
                sc.next();
            }
            o1=sc.nextDouble();
            System.out.print("Podaj ocenę 2:");
            while(! sc.hasNextDouble()){
                System.out.print("Blad, podaj poprawnie:");
                sc.next();
            }
            o2=sc.nextDouble();
        }
        while(!S.oceny(o1,o2));
        S.wypisz();
    }
}
class Student{
    String nazwisko;
    String imie;
    private double o1;
    private double o2;
    double srednia;
    boolean oceny(double oc1, double oc2){
        if(oc1<2){
            return false;
        }
        else if(oc1>5){
            return false;
        }
    }
}
```

```

        else if(oc2<2){
            return false;
        }
        else if(oc2>5){
            return false;
        }
        else{
            o1      = oc1;
            o2      = oc2;
            srednia = (o1+o2)/2;
            return true;
        }
    }
    Student(String n, String i){
        nazwisko = n;
        imie     = i;
    }
    void wypisz(){
        System.out.print("\nNazwisko: "+nazwisko);
        System.out.print("\nImię: "+imie);
        System.out.print("\nOcena 1: "+o1);
        System.out.print("\nOcena 2: "+o2);
        System.out.print("\nŚrednia: "+srednia);
    }
}

```

Wyniki:

```

Podaj ocenę 1:5
Podaj ocenę 2:4

Nazwisko: Faworyny
Imię: Franciszek
Ocena 1: 5.0
Ocena 2: 4.0
Średnia: 4.5
Process finished with exit code 0

```

5. Do klasy **Student2** dopisać przeciążoną (dwuargumentową) metodę **wstawOcene**, która pozwoli wstawić oceny w oba pola **ocena1**, **ocena2** w jednym wywoaniu i zwracającą wartość równą **false**., gdy którakolwiek z wprowadzonych ocen nie należy do przedziału [2,5], co powinno powodować ponowne wprowadzenie obu ocen.

Treść

Wyniki:

6. Utworzyć tablicę dwóch obiektów typu **Student2**, wczytać i wypisać ich pola.

Treść:

Wyniki:

7. Napisać klasę **Book** o polach **autor**, **tytul**, **wypożyczona** (typu **boolean**) posiadającą konstruktor dwuargumentowy nadający wartości polom klasy.

Treść:

Wyniki:

8. Napisać klasę **Czytelnik** z polami **nazwisko**, **Book1**, **Book2** (klasy **Book**), konstruktor jednoargumentowy nadający wartość polu **nazwisko**, metodę **wypożycz**, która sprawdza, czy czytelnik może wypożyczyć książkę (dopuszczalne jest wypożyczenie dwóch książek), jeśli tak, to umieszcza ją w polu **Book1** lub **Book2**), metodę **zwroc**. obsługującą zwrot książki.