

Sprawozdanie 2

Języki Programowania Obiektowego

Paweł Jamro, 292510, IMT

Kod opracowanych klas + komentarze

1. Klasa punkt:

```
public class Punkt
{
    protected float masa;

    //konstruktor domyślny
    public Punkt()
    {
        //przypisanie wartości domyślnej do masy
        this.masa = 1f;
    }

    //konstruktor z parametrami
    public Punkt(float masa)
    {
        // kontrola wartości
        if(masa > 0)
        {
            this.masa = masa;
        }

        else
        {
            this.masa = 1;
        }
    }

    //Główny moment bezwładności
    public float GłownyMomentBezwladnosci()
    {
        return 0;
    }

    //Moment bezwładności ze Steinera
    public float MomentBezwladnosciLiczonySteinerem(float odl)
    {
        return GłownyMomentBezwladnosci() + masa*odl*odl;
    }

    //metoda zwracająca masę punktu
    public void Opis()
    {
        System.out.println("Punkt Materialny");
    }

    //metoda, która pozwala zmienić masę punktu
    public void zmienianieMasy(float mass)
    {
        this.masa = mass;
    }

    //metoda, która pozwala zwrócić masę punktu
    public float zwracanieMasy()
    {
        return this.masa;
    }
}
```

2. Klasa pręt

```
public class Pret extends Punkt
{
    protected float dlugosc;
    //konstruktor domyślny
    public Pret() {

        //przypisuje stała wartosc z Superklasy
        super();
        //przypisuje stałe wartosci dlugosci
        this.dlugosc = 4f;
    }

    //konstruktor z parametrami
    public Pret( float masa, float dlugosc)
    { // kontrola wartosci, jesli bedzie minusowo przypisuje odgornie 1
        if(masa > 0) {
            this.masa = masa;
        }
        else {
            masa = 1;
        }
        if(dlugosc > 0) {
            this.dlugosc = dlugosc;
        }
        else {
            this.dlugosc = 1;
        }
    }

    //settery
    public void zmianaDlugosci (float dlugosc)
    {
        this.dlugosc = dlugosc;
    }

    //gettery
    public float zwrocDlugosc()
    {
        return this.dlugosc;
    }

    //przeciazana metoda licząca główny moment bezwładności
    public float GłównyMomentBezwładności()
    {
        return 1f/12f*masa*dlugosc*dlugosc;
    }

    //przeciazana metoda licząca moment bezwładności względem osi równoległej
    public float MomentBezwładnościSteiner(float odl)
    {
        return GłównyMomentBezwładności() + masa*odl*odl;
    }

    public void Opis()
    {
        System.out.println("Pret");
    }
}
```

3. Klasa kula:

```
public class Kula extends Punkt
{
    protected float promien;
    //konstruktor domyślny
    public Kula() {
        //przypisanie stałej wartości z Superklasy
        super();
        //przypisanie stałych wartości promienia
        this.promien = 1f;
        this.masa = 1f;
    }
    //konstruktor z parametrami
    public Kula( float masa, float promien)
    {
        if(masa > 0) {
            this.masa = masa;
        }
        else {
            this.masa = 1;
        }

        if(promien > 0) {
            this.promien = masa;
        }
        else {
            this.promien = 1;
        }
    }
    //setter
    public void zmianaPromienia(float promien) {
        this.promien = promien;
    }
    //getter
    public float zwrocPromien() {
        return this.promien;
    }
    //przeciążona metoda licząca główny moment bezwładności
    public float GłównyMomentBezwładności() {
        float moment = 2f/5f*masa*promien*promien;
        return moment;
    }
    //przeciążona metoda licząca moment bezwładności względem osi równoległej
    public float MomentBezwładnościSteiner(float odległość) {
        return GłównyMomentBezwładności() + masa*odległość*odległość;
    }
    public void Opis() {
        System.out.println("Kula");
    }
}
```

4. Klasa walec:

```
public class Walec extends Punkt
{
    protected float wysokosc;
    protected float promien;
    //konstruktor domyślny
    public Walec() {
        //przypisanie stałej wartości z Superklasy
        super();
        //przypisanie stałych wartości dla wysokosci i promienia
        this.wysokosc = 5f;
        this.promien = 5f;
    }
    //konstruktor z parametrami
    public Walec( float masa, float wysokosc, float promien)
    {
        // kontrola wartosci, jesli bedzie minusowo przypisuje odgornie 1;
        if(masa > 0) {
            this.masa = masa;
        }
        else {
            masa = 1;
        }

        if(wysokosc > 0) {
            this.wysokosc = wysokosc;
        }
        else {
            this.wysokosc = 1;
        }

        if(promien > 0) {
            this.promien = promien;
        }
        else {
            this.promien = 1;
        }
    }
    //setter
    public void zmianaWysokosci (float wysokosc) { this.wysokosc = wysokosc; }
    public void zmianaPromienia(float promien) { this.promien = promien; }
    //getter
    public float zwrocWysokosc() { return this.wysokosc; }
    public float zwrocPromien() { return this.promien; }
    //przebieżana metoda licząca główny moment bezwładności
    public float GłównyMomentBezwładności() { return masa*3f*promien*promien + 1f/12f*masa*wysokosc*wysokosc; }
    //przebieżana metoda licząca moment bezwładności względem osi równoległej
    public float MomentBezwładnościSteiner(float odleglosc)
    {
        return GłównyMomentBezwładności() + masa*odleglosc*odleglosc;
    }
    public void Opis() { System.out.println("Walec"); }
}
```

5. Program testowy:

```
package com.company;
import Bryly.*;
public class Main {

    public static void main(String[] args) {

        //1
        Walec walec = new Walec( masa: 3f, wysokosc: 4f, promien: 7f);
        Pret pret = new Pret( masa: 5f, dlugosc: 4f);
        Kula kula = new Kula( masa: 2f, promien: 7f);

        //2
        System.out.println("Podpunkt 2");
        System.out.println();
        walec.Opis();
        System.out.println("Masa walca = " + walec.zwracanieMasy());
        System.out.println("Promien walca = " + walec.zwrocPromien());
        System.out.println("Wysokosc walca = " + walec.zwrocWysokosc());
        System.out.println("Glowny moment walca = " + walec.GlownyMomentBezwladnosci());
        System.out.println("Moment wzledem osi przesuniętej o 4 dla walca = " + walec.MomentBezwladnosciSteiner( odleglosc: 4f));

        System.out.println();
        pret.Opis();
        System.out.println("Masa pręta = " + pret.zwracanieMasy());
        System.out.println("Długość pręta = " + pret.zwrocDlugosc());
        System.out.println("Glowny moment pręta = " + pret.GlownyMomentBezwladnosci());
        System.out.println("Moment wzledem osi przesuniętej o 4 dla pręta = " + pret.MomentBezwladnosciSteiner( odk: 4f));

        System.out.println();
        kula.Opis();
        System.out.println("Masa kuli = " + kula.zwracanieMasy());
        System.out.println("Promień kuli = " + kula.zwrocPromien());
        System.out.println("Glowny moment kuli = " + kula.GlownyMomentBezwladnosci());
        System.out.println("Moment wzledem osi przesuniętej o 4 dla kuli = " + kula.MomentBezwladnosciSteiner( odleglosc: 4f));

        //3
        Punkt[] tablica = new Punkt[3];
        tablica[0] = new Walec( masa: 3f, wysokosc: 4f, promien: 5f);
        tablica[1] = new Pret( masa: 4f, dlugosc: 2f);
        tablica[2] = new Kula( masa: 4f, promien: 6f);
        int counter = 0;

        // 4 i 5
        System.out.println();
        System.out.println("Podpunkt 4 i 5");
        System.out.println();
        for (Punkt p: tablica)
        {
            System.out.println("Obiekt nr. " + counter);
            System.out.println();
            p.Opis();
            System.out.println("Masa obiektu o numerze: " + counter+ " to: " + p.zwracanieMasy());
            if(p.getClass().getClassName() == "Bryly.Walec")
            {
                System.out.println("Promień obiektu o numerze: " + counter + " = " + ((Walec)p).zwrocPromien());
                System.out.println("Wysokosc obiektu o numerze: " + counter + " = " + ((Walec)p).zwrocWysokosc());
            }
            else if(p.getClass().getClassName() == "Bryly.Pret")
            {
                System.out.println("Promień obiektu o numerze: " + counter + " = " + ((Pret)p).zwrocDlugosc());
            }
            else if(p.getClass().getClassName() == "Bryly.Kula")
            {
                System.out.println("Promień obiektu o numerze: " + counter + " = " + ((Kula)p).zwrocPromien());
            }
            System.out.println("Glowny moment obiektu o numerze: " + counter+ " to: " + p.GlownyMomentBezwladnosci());
            System.out.println("Moment wzledem osi przesuniętej o 2 dla obiektu o numerze: " + counter+ " to: " + p.MomentBezwladnosciLiczonySteinerem( odk: 2f));
            counter++;
        }
    }
}
```

WYNIK PROGRAMU TESTOWEGO + OMÓWIENIE

Podpunkt 2

Walec

Masa walca = 3.0
Promień walca = 7.0
Wysokość walca = 4.0
Główny moment walca = 445.0
Moment względem osi przesuniętej o 4 dla walca = 493.0

Pręt

Masa pręta = 5.0
Długość pręta = 4.0
Główny moment pręta = 6.666667
Moment względem osi przesuniętej o 4 dla pręta = 86.666664

Kula

Masa kuli = 2.0
Promień kuli = 2.0
Główny moment kuli = 3.2
Moment względem osi przesuniętej o 4 dla kuli = 35.2

Podpunkt 4 i 5

Obiekt nr. 0

Walec

Masa obiektu o numerze: 0 to: 3.0
Promień obiektu o numerze: 0 = 5.0
Wysokość obiektu o numerze: 0 = 4.0
Główny moment obiektu o numerze: 0 to: 229.0
Moment względem osi przesuniętej o 2 dla obiektu o numerze: 0 to: 241.0

Obiekt nr. 1

Pręt

Masa obiektu o numerze: 1 to: 4.0
Promień obiektu o numerze: 1 = 2.0
Główny moment obiektu o numerze: 1 to: 1.3333334
Moment względem osi przesuniętej o 2 dla obiektu o numerze: 1 to: 17.333334

Obiekt nr. 2

Kula

Masa obiektu o numerze: 2 to: 4.0
Promień obiektu o numerze: 2 = 4.0
Główny moment obiektu o numerze: 2 to: 25.6
Moment względem osi przesuniętej o 2 dla obiektu o numerze: 2 to: 41.6

a)

```
Podpunkt 2

Walec
Masa walca = 3.0
Promień walca = 7.0
Wysokość walca = 4.0
Główny moment walca = 445.0
Moment względem osi przesuniętej o 4 dla walca = 493.0

Pret
Masa pręta = 5.0
Długość pręta = 4.0
Główny moment pręta = 6.666667
Moment względem osi przesuniętej o 4 dla pręta = 86.666664

Kula
Masa kuli = 2.0
Promień kuli = 2.0
Główny moment kuli = 3.2
Moment względem osi przesuniętej o 4 dla kuli = 35.2
```

Wyświetlenie dla wszystkich wybranych obiektów opisu, podstawowych danych oraz głównych momentów bezwładności, z twierdzenia Steinera dla osi odległej o 4.

b)

```
Obiekt nr. 0

Walec

Masa obiektu o numerze: 0 to: 3.0
Promień obiektu o numerze: 0 = 5.0
Wysokość obiektu o numerze: 0 = 4.0
Główny moment obiektu o numerze: 0 to: 229.0
Moment względem osi przesuniętej o 2 dla obiektu o numerze: 0 to: 241.0

Obiekt nr. 1

Pret

Masa obiektu o numerze: 1 to: 4.0
Promień obiektu o numerze: 1 = 2.0
Główny moment obiektu o numerze: 1 to: 1.3333334
Moment względem osi przesuniętej o 2 dla obiektu o numerze: 1 to: 17.333334

Obiekt nr. 2

Kula

Masa obiektu o numerze: 2 to: 4.0
Promień obiektu o numerze: 2 = 4.0
Główny moment obiektu o numerze: 2 to: 25.6
Moment względem osi przesuniętej o 2 dla obiektu o numerze: 2 to: 41.6
```

Wyświetlenie tych samych informacji co powyższej, ale dla każdego obiektu tablicy.

Odpowiedzi na pytania:

1. Zastąpienie pól prywatnych polami chronionymi w dziedziczeniu pozwala na dostęp klasie dziecko do tych pól. W przeciwnym wypadku klasa dziedzicząca nie będzie miała zezwolenia na korzystanie z pól oznaczonych modyfikatorem `private` z klasy rodzic.
2. Konstruktor superklasy (klasy rodzic) wywołuje się za pomocą konstrukcji: `super(parametry)`. Korzystamy z tego, gdy nie chcemy duplikować kodu na zainicjowanie pewnych zmiennych, które dziedziczymy po superklasie.
3. Dostęp do pól i metod superklasy możemy uzyskać poprzez dziedziczenie pod warunkiem, że modyfikatorem dostępu jest `protected` lub `public`. Wówczas odwołujemy się tak jakby to były metody i pola klasy dziedziczącej.
4. Przeciążanie metod pozwala nam utworzyć metody o takiej samej nazwie lecz przyjmujących różne argumenty. Ma to zastosowanie gdy chcemy zastosować pewną metodę lecz w zależności od potrzeb może ona przyjmować różne argumenty.