WSB Poznań WZ Chorzów
Kierunek Informatyka
Studia I stopnia
Rok I semestr 2

Programowanie obiektowe

Laboratorium 4

Zadanie Czworobok0

- Przekopiować kod z pliku czworobok.txt do pustego projektu i wykonać polecenia w nim zawarte

```
Odwołania: 0
public static void Main()
{
    // Rozwiaz problem z metoda Pole() trzema sposobami:

    // 1.Tworzac metode w klasie bazowej i stosujac polimorfizm

    // 2.Stosujac wlasciwosc GetType().Name i rzutowanie

    // 3.Stosujac metode abstrakcyjna w klasie bazowej

    Czworobok kw = new Kwadrat(5);
    Czworobok pr = new Prostokat(5, 4);

    Czworobok[] tab = { kw, pr };
    for (int i = 0; i < tab.Length; i++)
        Console.WriteLine(tab[i].Pole());
}
```

Zadanie Czworobok

- Utworzyć klasę abstrakcyjną **Czworobok** z dwoma polami **x** oraz **y**, jednym konstruktorem dwuparametrycznym do inicjowania pól oraz metodę abstrakcyjną **Pole**

- Utworzyć klasę **Kwadrat** dziedziczącą po klasie **Czworobok** z konstruktorem parametrycznym opartym na konstruktorze bazowym.

- W klasie **Kwadrat** utworzyć metodę przesłaniającą metodę **Pole** z własną implementacją obliczania pola kwadratu

- Utworzyć klasę **Prostokat** dziedziczącą po klasie **Czworobok** z konstruktorem dwuparametrycznym opartym na konstruktorze bazowym.

- W klasie **Prostokat** utworzyć metodę przesłaniającą metodę **Pole** z własną implementacją obliczania pola prostokąta

Zadanie Czworobok cd.

- Utworzyć klasę **Trapez** dziedziczącą po klasie **Czworobok** z trzema polami **podst_a**, **podst_b**, **wysokosc** oraz z konstruktorem trzyparametrycznym opartym na konstruktorze bazowym

- W klasie **Trapez** utworzyć metodę przesłaniającą metodę **Pole** z własną implementacją obliczania pola trapezu

- W metodzie **Main** utwórz tablicę obiektów klasy **Czworobok** z jednym kwadratem, trójkątem oraz trapezem.

- W pętli przetworzyć tablicę wywołując metodę **Pole** na każdym elemencie tablicy.

Zadanie Figury2

- Utworzyć interfejs **IFigura** z jedną właściwością **obwod** oraz z jedną metodą **Pole**

- Utworzyć klasę **Kwadrat** implementującą właściwość **obwód** jako obwód kwadratu (akcesor **get**) oraz metodę **Pole** jako pole kwadratu. W klasie ma być utworzony konstruktor parametryczny inicjujący pole **bok_a**

- Utworzyć klasę **Trojkat** implementującą właściwość **obwód** jako obwód trójkąta (akcesor **get**) oraz metodę **Pole** jako pole trójkąta (wzór Herona). W klasie ma być utworzony konstruktor parametryczny inicjujący pola **bok_a, bok_b, bok_c**

- W klasie głównej utworzyć dwa obiekty: kwadrat oraz trójkąt i przetestować operacje na obiektach

Zadanie Ludolfina (pi)

- Utworzyć dwa interfejsy **ILeibniz** oraz **IEuler** z jedną metodą **Ludolfina** pobierającej argument typu **int**
- Utworzyć klasę **Obliczenia** dziedziczące oba interfejsy
- Zaimplementować metodę **Ludolfina** w klasie **Obliczenia** raz stosując **wzór Leibniza**, a drugi raz **Eulera**

niemiecki matematyk Gottfried Wilhelm Leibniz (XVII w.): $\frac{\pi}{4} = 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \frac{1}{9} + \ldots$

szwajcarski matematyk Leonhard Euler (XVIII w.): $\frac{\pi^2}{6} = 1 + \frac{1}{2^2} + \frac{1}{3^2} + \frac{1}{4^2} + \frac{1}{5^2} + \ldots$

Zadanie **Ludolfina (pi) cd.**

- W metodzie **Main** klasy podstawowej utworzyć obiekt **ob1** klasy **Obliczenia**
- Utworzyć obiekt **mat1** interfejsu **ILeibniz** i dokonać rzutowania na niego obiektu **ob1**
- Wywołać metodę **Ludolfina** na obiekcie **mat1**
- W ten sam sposób przećwiczyć wywołanie metody **Ludolfina** na interfejsie **IEuler**
- Porównać wyniki metody **Ludolfina** dla **n=20** wyświetlając różnice względem **Math.PI**

Zadanie Czlowiek3

```
interface ICzlowiek1
{
    Odwołania: 2
    void Idz(int x);          // Rozpoczyna od marszu z zadaną prędkością
    1 odwołanie
    void Biegnij(int x);      // Rozpoczyna od biegu z zadaną prędkością
    Odwołania: 2
    void Biegnij();           // Zaczyna biec po początkowym marszu zwiększając
                              // aktualną prędkość dwukrotnie
}
1 odwołanie
interface ICzlowiek2
{
    Odwołania: 3
    void Przyspiesz();        // Przyspiesza bieg o 2 km. lub marsz o 1 km.
    Odwołania: 5
    void Zwolnij();           // Zwalnia do połowy aktualnej prędkości lub
                              // Spaceruje/stoi przy prędkosci <1 km.
}
```

- Utworzyć dwa interfejsy oraz klasę pochodną, a następnie stworzyć obiekt i przetestować program.

Zadanie Czlowiek3 cd.