

## POJ

### Laboratorium 3

#### Zadania do samodzielnego wykonania :

##### Zadanie 1:

Stwórz klasę kwadrat. Umieść w niej pole bok, a także metody do obliczania obwodu i pola. Dodaj konstruktor.

Stwórz instancję i wywołaj metody obiektu.

##### Zadanie 2:

Stwórz klasę Punkt która reprezentować będzie punkt w przestrzeni x i y. Ma posiadać:

- pola x i y
- konstruktor bezargumentowy tworzący punkt w koordynatach (0,0)
- konstruktor pozwalający stworzyć punkt o określonych koordynatach
- metodę pozwalającą na obliczenie odległości między tym punktem a innym punktem podanym jako x i y
- metodę pozwalającą na obliczenie odległości między tym punktem a innym punktem podanym jako obiekt typu Punkt

##### Zadanie 3:

Stwórz klasę TV. Ma ona mieć pola:

- channel – reprezentujące ustawiony kanał
- volume – reprezentujące ustawioną głośność
- on – reprezentujące to czy TV jest włączone

Stwórz klasę Remote (pilota od telewizora). Ma ona mieć pole TV (typu obiektowego TV).

Dodaj metody do klasy **Remote** które mogą:

zmieniać kanał w górę / w dół ( w przedziale 1-20)

zmieniać głośność w górę / w dół (w przedziale 1-10) włączać / wyłączać telewizor

Pamiętaj, że można zmieniać kanały / głośność tylko wtedy, gdy TV jest włączony! Stwórz instancje TV, pilota, i pokaż, że pilot działa.

##### Zadanie 4:

Stwórz klasę Wiatrak. Klasa ma posiadać:

- trzy stałe o nazwie SLOW, MEDIUM i FAST o wartościach 1, 2 i 3 odpowiednio
- pole *speed* które przechowuje obecną prędkość wiatraka
- pole *on* które przechowuje stan wiatraka (włączony lub nie)
- pole *radius* które przechowuje promień wiatraka
- pole *color* które przechowuje kolor wiatraka
- konstruktor bezargumentowy który tworzy domyślny wiatrak
- konstruktor argumentowy
- metodę „informacje” która wyświetli w jednym Stringu wszystkie informacje o wiatraku

#### Zadanie 5:

Stwórz klasę konto. Konto ma mieć swoje id i ma przechowywać ilość pieniędzy na koncie. Ma umożliwiać sprawdzenie ilości pieniędzy na koncie, wypłacenie i wpłacenie środków. Stwórz tablicę 10 kont, nadaj każdemu z nich unikalne id i wartość początkową środków: 100 Stwórz klasę bankomat, który wyświetli menu dla użytkownika: ([ENTER] dla celów poglądowych)  
Enter an id: 4 [ENTER]

```
Main menu
1: check balance
2: withdraw
3: deposit
4: exit
Enter a choice: 1 [ENTER]
The balance is 100.0
```

Program ma umożliwiać wykonywanie operacji 1, 2 i 3 bez wychodzenia z programu albo bez ponownego podawania ID (siedzimy na danym id, wykonujemy operacje, jak mamy dość, to wpisujemy EXIT)

Gdy użytkownik wybierze EXIT, program ma zapytać znowu o id (nie ma się wyłączyć).

Stwórz instancje i wywołaj metody obiektów.

#### Zadanie 6:

Stwórz symulator walki – stwórz klasę Gracz i Przeciwnik. Każdy z nich ma punkty życia, broń oraz zbroję. Niech użytkownik może wybrać jedną z dostępnych broni / zbroi, a następnie może wybrać przeciwnika z którym walczy. Walka toczy się cios za cios, najpierw gracz, potem przeciwnik i tak aż do śmierci jednego z nich. Postarać się zrobić tak, by każdy z nich miał szansę wygrać (niekoniecznie równą, np. podyktowaną wyborem ekwipunku). Można dodać coś od siebie jak ktoś ma wenę.

Rozwiązanie zadań powinno być udostępnione w na indywidualnym repozytorium github studenta maksymalnie do końca czwartego dnia po zajęciach:

- zajęcia w poniedziałek -> termin do czwartku 23:59:59
- zajęcia w środę -> termin do niedzieli 23:59:59
- zajęcia w czwartek -> termin do poniedziałku 23:59:59

Zadania powinny być zrealizowane w osobnych klasach w ramach jednego projektu, a umieszczony w **prywatnym** repozytorium,