

# Laboratorium 10 - Przeciążenie

## Składając wszystko razem...

1. Proszę zaimplementować klasę implementującą wektor w przestrzeni  $R^2$  (nazwa: `wektor2D`) według następujących wytycznych. Kod proszę pisać krok po kroku kompilując i sprawdzając działanie kodu po każdym kroku.
2. Proszę stworzyć klasę z prywatnymi zmiennymi przechowującymi współrzędne wektora oraz konstruktorem. Proszę przetestować klasę wywołując konstruktor z funkcji `main()`. Rozwiązanie proszę zapisać jako plik `Lab10_1_solved.cpp` i umieścić w swojej gałęzi git.
3. Proszę rozszerzyć klasę o gettery i settery. Proszę przetestować klasę wywołując konstruktor z funkcji `main()`. Rozwiązanie proszę zapisać jako plik `Lab10_2_solved.cpp` i umieścić w swojej gałęzi git.
4. Proszę skonstruować klasę dziedziczącą po klasie `wektor2D` (nazwa: `wektor2Dpro`) rozszerzającą jej funkcjonalność o iloczyn wektora i skłara. Proszę przetestować klasę wywołując metodę z funkcji `main()`. Rozwiązanie proszę zapisać jako plik `Lab10_3_solved.cpp` i umieścić w swojej gałęzi git.
5. Proszę rozszerzyć klasę `wektor2Dpro` o przeciążony operator dodawania wektorów. Proszę przetestować klasę wywołując operator z funkcji `main()`. Rozwiązanie proszę zapisać jako plik `Lab10_4_solved.cpp` i umieścić w swojej gałęzi git.

Jeśli (zgodnie z zasadami dobrego programowania) składowe klasy `wektor2D` są prywatne to być może niezbędne będzie zadeklarowanie getterów w `wektor2d` jako funkcji stałych. Więcej: <https://stackoverflow.com/questions/5973427/error-passing-xxx-as-this-argument-of-xxx-discards-qualifiers>

```
int wektor2d::get_y(void) const{
    return y;
}
```

6. Proszę rozszerzyć klasę `wektor2D` o przeciążony operator strumienia (`<<`) używając funkcji zaprzyjaźnionej (`friend`). Proszę przetestować klasę wywołując operator z funkcji `main()`. Rozwiązanie proszę zapisać jako plik `Lab10_5_solved.cpp` i umieścić w swojej gałęzi git.

## Złożone struktury danych

1. Proszę zaimplementować listę jednokierunkową przechowującą następujące dane `imie`, `nazwisko`, `nr_indeksu`.
2. Proszę zaimplementować funkcję tworzącą listę o zadanej długości
3. Proszę zaimplementować funkcję wypełniającą listę danymi. `imie=Jan`, `nazwisko=Kowalski`, `nr_indeksu` - kolejna liczba.
4. Proszę zaimplementować funkcję wyświetlania zawartości całej listy
5. Proszę zaimplementować funkcję dodawania elementu na końcu listy
6. Proszę zaimplementować funkcję usuwania wskazanego elementu listy
7. Proszę zaimplementować funkcję usuwającą listę (!czyszczenie pamięci!)
8. DLA AMBITNYCH: Proszę zmodyfikować implementację listy i funkcji tak, aby lista była dwukierunkowa