

Języki Programowania, Zadanie 1

(1) Należy otworzyć pusty plik tekstowy - powinien mieć rozszerzenie cpp. Należy dołączyć biblioteki wejścia/wyjścia:

```
#include <iostream>
```

zdefiniować przestrzeń nazw:

```
using namespace std;
```

oraz napisać funkcję `int main()`. W funkcji wypisać na ekran napis „Punkt [3,4]”.

Skompilować i uruchomić. *Pamiętajcie, żeby regularnie zapisywać i kompilować wasze programy!*

Wczytać dwie liczby z klawiatury. Wypisać je na ekran podobnie jak powyższe „3” i „4”.

(2) Należy stworzyć strukturę **Punkt** reprezentującą punkt w przestrzeni dwuwymiarowej. Struktura ta powinna zawierać dwa pola typu double (składowe x i y) :

```
- double x;
```

```
- double y;
```

oraz zawierać następujące metody (funkcje składowe):

```
- void zapisz(double xx, double yy) - umożliwiającą zapisanie danego punktu
```

```
- void wypisz() - wypisującą dany punkt na ekran w formacie [x,y]
```

Punkt
-x
-y
+zapisz(xx, yy)
+wypisz()
+odleglosc(xx, yy)
+dodaj(xx, yy)
+kopiuj(Punkt p)

W programie głównym należy stworzyć pojedynczy obiekt typu Punkt ($x = 1, y = 4$) i wypisać go na ekran. Następnie wczytać dwie liczby z klawiatury, odpowiadające x oraz y i stworzyć drugi punkt z podanymi wartościami, również wypisać go na ekran.

(3) Proszę przerobić strukturę na klasę. Słowo kluczowe „struct” zamienić na słowo kluczowe „class”
Wszystkie składniki i metody mogą być publiczne (public).

- **Proszę sprawdzić, jaki efekt ma NIE umieszczenie słowa kluczowego “public” w kodzie klasy. Proszę efekt zapisać w komentarzu w kodzie.**

(4) Dopisać funkcję składową **zwracającą** odległość punktu na którym wywołana jest metoda od punktu zadanego jako argument funkcji

[**typ zwracany**] `odleglosc(double X, double Y)` | by użyć pierwiastka należy załączyć bibliotekę math.h:

```
#include <math.h>
```

Przetestować ją w funkcji main() wypisując na ekran odległość pierwszego punktu od punktu [5,6].

UWAGA! Jeśli w treści zadania pojawia się prośba o napisanie funkcji która „zwraca” wartość, funkcja powinna tę wartość zwracać (poprzez return bądź parametr, a nie jedynie wypisywać na ekran!)

(5) Klasa powinna zostać podzielona na dwa oddzielne pliki: `punkt.h` i `punkt.cpp`.

W pliku *.h powinna się znaleźć definicja klasy (słowo kluczowe class, składniki klasy, oraz deklaracje funkcji), natomiast w pliku *.cpp – ciało funkcji składowych (metod). Dodatkowo należy stworzyć główny plik programu zawierający funkcję main() - razem 3 pliki.

(6) Dopisać następującą funkcję składową:

- *dodając wielkości liczbowe = zwiększające wartości współrzędnych punktu:*

```
void dodaj(double xx, double yy)
```

- *przypisując wartości zadanego punktu (zmieniający wartości x i y punktu na którym funkcja jest wywoływana) :*

```
void kopiuj(Punkt p)
```

Przetestować obie w funkcji main() wypisując na ekran punkt pierwszy po dodaniu do niego [2,2] i punkt drugi, po przypisania mu wartości z punktu pierwszego.

Wskazówki:

- By dostać się do składników struktury „na zewnątrz” używamy ".", np. `cout<<A.F()<<endl;` (jeśli są publiczne)
- Należy pamiętać o załączeniu biblioteki punkt.h w plikach *.cpp (`#include "punkt.h"`).
- W pliku punkt.cpp przy definicjach funkcji należy pamiętać o dodaniu operatora zakresu `"punkt : :"`