

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 „(3,4)”. Skompilować (używając linii komend).

```
g++ -Wall program.cpp -o program
```

Wczytać dwie liczby z klawiatury. Wypisać je na ekran zamiast „3” i „4”. (0.5 p)

(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 (1 p)

- `void wypisz()` - wypisującą dany punkt na ekran w formacie `[x,y]` (0.5 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. (1 p)

Zgłosić ukończenie pierwszej części zadania prowadzącemu [łącznie: 3 pkt].

(3) Proszę przerobić strukturę na klasę. Słowo kluczowe „`struct`” zamienić na słowo kluczowe „`class`”) (0.5 p)

Wszystkie składniki i metody mogą być publiczne (`public`).

- Proszę sprawdzić, jaki efekt ma NIE umieszczenie słowa kluczowego „`public`” w kodzie klasy.

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

```
odleglosc(int X, int 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). (0.5 p)

(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. (0.5 p)

```
g++ -Wall Punkt.cpp program.cpp -o program
```

(6) Dopisać następującą funkcję składową:

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

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

Przetestować ją w funkcji `main()` wypisując na ekran punkt pierwszy po dodaniu do niego `[2,2]` (0.5 p)

Wskazówki:

- By dostać się do składników struktury „na zewnątrz” używamy „`.`”, np. `cout<<A.x<<endl;`

- 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 : :"`

Wszystkie napisane funkcje muszą zostać przetestowane w programie głównym.