

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 „[1,2]”. Skompilować (używając linii komend).

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

Wczytać dwie liczby z klawiatury. Wypisać je na ekran zamiast „1” i „2”. (0.5 p)

(2) Należy stworzyć strukturę Wektor reprezentującą wektor dwuwymiarowy. 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 wektora (1 p)
```

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

W programie głównym należy stworzyć pojedynczy obiekt typu Wektor ($x = 2$, $y = 3$) i wypisać go na ekran.

Następnie wczytać dwie liczby z klawiatury, odpowiadające x oraz y i stworzyć drugi Wektor 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 długość wektora

```
dlugosc() | by użyć pierwiastka należy załączyć bibliotekę math.h: #include <math.h>
```

Przetestować ją w funkcji main() wypisując na ekran długość pierwszego i drugiego wektora. (0.5 p)

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

W pliku *.h powinna się znaleźć definicja klasy (słowo kluczowe class, składniki klasy, oraz deklaracje funkcji), natomiast w pliku *.cpp – ciała funkcji składowych (metod). Dodatkowo należy stworzyć główny plik programu zawierający funkcję main() - razem 3 pliki. (0.5 p)

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

(6) Dopisać następujące funkcje składowe (0.5 p):

- dodający wielkości liczbowe zwiększające długości wektora: `void dodaj(double xx, double yy)`

- odejmujący zadany wektor (zmieniający wartości x i y wektora na którym funkcja jest wywoływana) :

```
void odejmij(Wektor w)
```

Przetestować obie w funkcji main() wypisując na ekran wektor (1) pierwszy po dodaniu do niego [2,2]

(2) drugi, po odjęciu do niego wektora pierwszego

Wskazówki:

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

- Należy pamiętać o załączeniu biblioteki wektor.h w plikach *.cpp (`#include "wektor.h"`).

- W pliku wektor.cpp przy definicjach funkcji należy pamiętać o dodaniu operatora zakresu "wektor : :"

Wszystkie napisane funkcje muszą zostać przetestowane w programie głównym.

Zadanie dodatkowe

Należy dopisać funkcje zewnętrzne:

- `wektor dodaj(wektor w1, wektor w2)` (0.25 p)

- wektor odejmij (wektor w1, wektor w2) **(0.25 p)**