

Zadanie 3, Języki Programowania, 23.10.2012

Proszę stworzyć program umożliwiający zapamiętywanie zbioru prostokątów (ich położenia na płaszczyźnie oraz promieni / długości boków). Następnie możliwe powinno być sprawdzenie, czy podany punkt o współrzędnych (x,y) znajduje się wewnątrz którejkolwiek z figur. Wszystkie podawane współrzędne mają być dodatnie.

Cały program należy napisać w dwóch plikach: nagłówkowym (*.h) oraz implementacyjnym, zawierającym także funkcję główną (*.cpp).

Należy stworzyć klasy:

Punkt (0.5 p)

Pola publiczne: • double x,y; // (x,y) – współrzędne punktu

Metody publiczne: • void Zapisz(double X, double Y),
• void Wypisz()

Domyślnymi wartościami dla współrzędnych punktu powinno być $x = y = 30$.

Prostokat (1.0 p)

Pola prywatne: • Punkt p, double a,b; // (p) – wybrany róg prostokąta, a,b – długości boków

Metody publiczne: • void Zapisz(double X, double Y, double A, double B),
• void Wypisz(),
• double Pole() //zwraca pole powierzchni
• metody set i get do ustawiania i pobierania pól prywatnych
np. void SetA(double A), double GetX()

Domyślnymi wartościami dla wymiarów prostokąta powinny być $A = B = 30$.

W funkcji głównej należy stworzyć pojedyncze obiekty Punkt (x=3, y=3), Prostokat (x=4, y=10) i wypisać je na ekran. Ponadto wypisać na ekranie pole prostokąta, następnie zgłosić ukończenie części zadania prowadzącemu. Należy podobnie testować kod i zgłaszać się po każdej napisanej funkcji/części.

Należy przeciążyć operator dodawania (operator+)(0.5 p)

Dla klasy prostokąt (zwracający prostokąt o współrzędnych pierwszego z podanych prostokątów i długości boków będących sumą długości boków prostokątów dodawanych).

Prostokat operator+(Prostokat p1, Prostokat p2);

Operator należy zaprzyjaźnić z klasą Prostokat, oraz odwoływać się bezpośrednio do prywatnych pól składowych klasy.

Ponadto należy stworzyć funkcje (nie funkcje składowe!): (1.5 p)

Prostokat SkopiujProstokat(Prostokat &p) – zwracającej nowy prostokąt poprzez kopiowanie długości boków prostokąta p, ale zerując współrzędne x oraz y. **Do czego tutaj służy „&”?**

Prostokat* SkopiujProstokatWsk(Prostokat *p) – zwracającej nowy prostokąt poprzez kopiowanie długości boków prostokąta p, zerując współrzędne x oraz y, ale działającą na wskaźnikach.

void ZmienProstokat(Prostokat &p, double x, double y, double a, double b) – zmieniającą wartości (a,b,x,y) istniejącego obiektu Prostokat p.

Funkcje (z tymi samymi nazwami – przeciążanie nazw): (1,5 p)

bool Wewnatrz (Prostokat P, double x, double y) – zwraca „true” jeśli punkt (x,y) znajduje się wewnątrz prostokąta P.

bool Wewnatrz (Prostokat P, Punkt pkt)– zwraca „true” jeśli punkt (pkt) znajduje się wewnątrz prostokąta P.

bool Wewnatrz(Prostokat *P, double x, double y) – zwraca „true” jeśli punkt (x,y) znajduje się wewnątrz **któregokolwiek** z prostokątów zapisanych w podanych tablicach.

Działanie wszystkich funkcji powinno zostać przetestowane dla przykładowych obiektów w funkcji main.

Zadanie dodatkowe:

1. double* Boki() zwracającą tablicę dwóch liczb: na pierwszym

miejscu długość boku a, na drugim miejscu tablicy: długość boku b

2. W przyszłości chcielibyśmy nasze figury wyświetlać na ekranie. By prezentowały się odpowiednio będziemy, prócz współrzędnych i wymiarów, przyporządkowywać im kolor. Lista dostępnych kolorów to: czerwony, zielony, niebieski, pomarańczowy.

Proszę stworzyć typ wyliczeniowy przechowujący informację o kolorze dla klasy prostokąt, oraz dodać dodatkowy składnik: kolor do tej klasy.

Szablon dla zawartości funkcji głównej (main):

```
//utworzenie pojedynczego punktu (x=3,y=3)
```

```
//utworzenie pojedynczego prostokąta p1 (x=4,y=10) - a, b: domyślne
```

```
//wypisanie punktu na ekran
```

```
//wypisanie prostokąta na ekran
```

```
//wypisanie pola prostokąta na ekran
```

```
---1.5 pkt---
```

```
//stworzenie drugiego prostokąta p2 (x=1, y =10, a = 10, b = 5)
```

```
//stworzenie trzeciego prostokąta = p1 + p2
```

```
//wypisanie trzeciego prostokąta na ekran
```

```
---2 pkt---
```

```
//utworzyć prostokąt p3 kopiując prostokąt p2 przy użyciu funkcji SkopiujProstokat
```

```
//utworzyć wkaźnik na prostokąt p4 kopiując prostokąt p3 przy użyciu funkcji
```

```
SkopiujProstokatWsk. Wypisać prostokąt p4 na ekran.
```

```
//zmienić prostokąt p3 tak, by jego składowe wynosiły (1,1,1,1), wypisać na ekran prostokąt p3
```

```
//wypisać na ekran boki prostokąta p4 uzyskane przy użyciu funkcji Boki()
```

```
---3.5 pkt---
```

```
//sprawdzenie funkcji Wewnatrz (wypisanie na ekranie "0" lub "1") dla prostokąta p1 i punktów: (1,1) oraz (5,11) - zapisanych jako dwie wielkości typu double
```

```
//sprawdzenie funkcji Wewnatrz (wypisanie na ekranie "0" lub "1") dla prostokąta p1 i punktów: (2,2) oraz (6,12) - zapisanych jako obiekty typu Punkt
```

```
//utworzenie 3-elementowej tablicy prostokątów, wpisanie w kolejne komórki prostokątów p1, p2, p3.
```

```
// sprawdzenie funkcji Wewnatrz (wypisanie na ekranie "0" lub "1") dla tablicy prostokątów
```

```
---5 pkt----
```