

Języki Programowania, Czwartek 16.10.2013, 16:15-17:45

Zadanie 2

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

Cały program należy napisać w pięciu plikach

Należy stworzyć klasę:

Prostokat (1.0 p)

Pola prywatne: double x,y,a,b; //(x,y) – wybrany róg prostokąta, a,b – długości boków
Metody publiczne: Zapisz(double X, double Y, double A, double B), Wypisz(),
double Pole() //zwraca pole powierzchni
oraz metody set i get do ustawiania i pobierania pól prywatnych
Domyślnymi wartościami dla wymiarów prostokąta powinny być $A = B = 30$.

W funkcji głównej należy stworzyć obiekt Prostokat (x=4, y=10), wypisać go na ekran oraz wypisać na ekranie jego pole, 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 prostokat (zwracający prostokat o współrzędnych pierwszego z podanych prostokątów i długości boków będących sumą długości boków prostokątów dodawanych). **Operator należy zaprzyjaźnić z klasą Prostokat, oraz odwoływać się bezpośrednio do prywatnych pól składowych klasy.**

Oraz napisać klasę okrag:

Okrag (1.0 p)

Pola prywatne: double x,y,R; //(x,y) – środek okręgu, R- promień
Metody publiczne: Zapisz(double X, double Y, double r), Wypisz(), double Pole()
metody set i get do ustawiania i pobierania pól prywatnych,
składową x klasy Okrag, na drugim miejscu tablicy: składową y.
Domyślną wartością dla promienia okręgu powinno być 10.

Funkcje (z tymi samymi nazwami – przeciążanie nazw): (1.0 p)

bool Wewnatrz – argumentami funkcji powinien być obiekt klasy Prostokat P, oraz współrzędne x oraz y, zwraca „true” jeśli punkt (x,y) znajduje się wewnątrz prostokąta P.

bool Wewnatrz – argumentami funkcji powinien być obiekt klasy Okrag O, oraz współrzędne x oraz y, zwraca „true” jeśli punkt (x,y) znajduje się wewnątrz okręgu O.

bool Wewnatrz – argumentami funkcji powinny być dwie tablice: prostokątów i okręgów, ilość prostokątów w tablicy, ilość okręgów w tablicy, oraz współrzędne punktu x, y, zwraca „true” jeśli punkt (x,y) znajduje się wewnątrz **któregokolwiek** z prostokątów bądź okręgów zapisanych w podanych tablicach.

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

Szablon dla zawartości funkcji głównej (main):

```
//utworzenie pojedynczego prostokąta
//wypisanie prostokąta oraz jego pola      Prostokat p1:(4,10,30,30), pole: 900
--- 1 pkt ---
//utworzyć prostokat p2 (9,9,5,7)
//utworzyć prostokat p3 dodając prostokaty p1 oraz p2
Prostokat p3:(4,10,35,37)
--- 2 pkt ---
//utworzenie pojedynczego okręgu
Okrag o1:(3,3,10), pole: 62.8
--- 3 pkt ---
//utworzyć tablice (a) dwóch okręgów i (b) trzech prostokątów
//jako pierwszy prostokąt wpisać obiekt p1, jako drugi: p2 jako trzeci (2,3)
//jako pierwszy okrąg wpisać obiekt o1, jako drugi okrąg: (13,14,5)
--- 4 pkt ---
//wypróbować funkcje Wewnatrz na utworzonych tablicach figur
Wewnatrz - Punkt (1,1):0 Punkt (20,20):1 //testowanie funkcji Wewnatrz
Wewnatrz - Punkt (0,0):1 Punkt (3.5,15):0 //testowanie funkcji Wewnatrz
--- 5 pkt ---
```

Zadanie dodatkowe:

Stworzyć funkcje (nie funkcje składowe!):

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

void ZmienProstokat(Prostokat *p, double X, double Y, double A, double B) – zmieniającą wartości (a,b,x,y) istniejącego obiektu Prostokat.

Stworzyć funkcje (nie funkcje składowe!):

Okrag* SkopiujOkrag(Okrag *o) – zwracającej nowy okrąg poprzez kopiowanie promienia okręgu o, zerując współrzędne x oraz y, ale działającą na wskaźnikach.

2W 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.

double* Boki() zwracającą tablicę dwóch liczb: na pierwszym miejscu długość boku a, na drugim miejscu tablicy: długość boku b