Autor: Dariusz Konieczny

Wersja draft 1.0

Programowanie obiektowe - zadanie 2

------------------------------------------

Oprogramować klasę CPoint2D, zapamiętującą punkt w przestrzeni dwuwymiarowej. Dla celów tej listy zadań składowe mają zostać zapamiętane jako *dwa wskaźniki na typ* **double**. Stworzyć konstruktory: kopiujący i z dwoma wartościami typu double. NIE ma być konstruktora domyślnego! Dopisać gettery i settery.

Stworzyć klasę CRectangle, do pamiętania prostokąta o bokach równoległych do osi współrzędnych, która używając wcześniej napisanej klasy pamięta dwa naprzeciwległe wierzchołki. Dla tej klasy stworzyć trzy kontruktory: jeden otrzymujący 4 liczby typu doble, drugi otrzymujący dwa punkty i trzeci - kopiujący.

Dodatkowo klasa CRectangle ma posiadać metodę obliczającą pole prostokąta oraz metodę wypisująca informacje o prostokącie w formacie:

CRectangle(CPoint(x1,y1),CPoint(x2,y2))

W programie testującym mamy dwie proste tablice wskaźników: punktów i prostokątów. Wielkość obu tablic ustalana jest na początku programu odpowiednią komendą.

Komendy programu:

Wykonanie każdej komendy zaczyna się wypisaniem znaku wykrzyknika (!) oraz powtórzeniem komendy. Potem wypisywana jest dodatkowa informacje zależna od komendy.

# go <ilePunktow> <ileProstokatow> == createDefs <ileElementow> ...

///////////////////////////////////////// PUNKT ////////////////////////////////////////////////////////////////////////////////

Stworzenie tablicy punktów o długości <ilePunktow> oraz tablicy prostokątów o długości <ileProstokatow>. Po skończeniu wypisanie w jednej linie napisu „DONE”, w przypadku niepoprawnych danych „ERROR”.

W kolejnych opisach takie wypisywanie odpowiedzi będzie skrótowo opisane jako

DONE/ERROR.

# createPoint <nrPozycji> <x1> <y1> == create

Stworzenie w pozycji <nrPozycji> punktu za pomocą konstruktora z danych x1, y1.

DONE/ERROR

# createPointCopy <nrPozycji> <nrPozycjiSkad> == createCopy

Stworzenie w pozycji <nrPozycji> punktu za pomocą konstruktora kopiującego z pozycji <nrPozycjiSkad>. DONE/ERROR

# setPoint <nrPozycji> <x1> <y1> == setValue

Ustawienie wartości współrzędnych dla punktu z <nrPozycji>. DONE/ERROR

getPointX <nrPozycji> == getVal

getPointY <nrPozycji> == getVal

Wypisanie odpowiedniej wartości punktu z <nrPozycji>. Jeśli niemożliwe to ERROR.

///////////////////////////////// PROSTOKAT ///////////////////////////////////////////////////////////////////

# createRectDouble <nrPozycji> <x1> <y1> <x2> <y2> == create

Stworzenie w pozycji <nrPozycji> prostokąta za pomocą konstruktora z danych x1, y1,x2,y2.

DONE/ERROR

# createRectPoints <nrPozycji> <nrPunktu1> <nrPunktu2> == create

Stworzenie w pozycji <nrPozycji> prostokąta za pomocą konstruktora z użyciem punktów z pozycji <nrPunktu1> <nrPunktu2>. DONE/ERROR

# createRectCopy <nrPozycji> <nrPozycjiSkad> == createCopy

Stworzenie w pozycji <nrPozycji> prostokąta za pomocą konstruktora kopiującego z pozycji < nrPozycjiSkad >. DONE/ERROR

fieldRect <nrPozycji> == print

Wypisanie wartości pola prostokąta z pozycji <nrPozycji>. Jeśli niemożliwe to ERROR.

# showRect <nrPozycji> == print

Wypisanie informacji o prostokącie wg formatu z zadania. Jeśli niemożliwe to ERROR.