

# kurs języka C++

## kolory i punkty

Instytut Informatyki  
Uniwersytetu Wrocławskiego

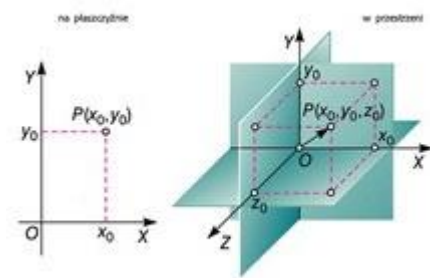
Paweł Rzechonek

### Prolog

Kartezjański układ współrzędnych w przestrzeni  $n$ -wymiarowej to uporządkowany ciąg  $n$  liczb opisujący położenie punktu w tej przestrzeni.

W przypadku płaszczyzny są dwie wzajemnie prostopadłe osie liczbowe przecinające się w punkcie  $O$ : jedna (zwykle pozioma) nazywa się osią odciętych a druga (zwykle pionowa) osią rzędnych. Każdemu punktowi  $P$  tej płaszczyzny przypisuje się parę liczb  $(x_P, y_P)$  takich, że  $x_P$  i  $y_P$  są rzutami prostokątnymi odpowiednio na oś  $OX$  i  $OY$ .

W przypadku przestrzeni są trzy wzajemnie prostopadłe osie liczbowe. Każdemu punktowi  $Q$  tej przestrzeni przypisuje się parę liczb  $(x_Q, y_Q, z_Q)$ .



Prostokątny układ współrzędnych nazywa się kartezjańskim, ponieważ nazwa ta pochodzi od Kartezjusza (Rene Descartes, 1596 - 1650) znakomitego matematyka, przyrodnika, filozofa, który był prekursorem geometrii analitycznej i zaprezentował tą ideę w 1637 roku w traktacie *La Géométrie*.

### Zadanie

Zdefiniuj **klasę kolor** reprezentującą kolor zapisany w systemie RGB. W klasie tej umieść **trzy pola**, które będą pamiętać nasycenie daną barwą, czyli wartości z zakresu 0...255. Zdefiniuj dwa **konstruktory**: **jeden bezargumentowy**, który będzie ustawiać **kolor na czarny (same zera)** oraz drugi, który zainicjalizuje wartości poszczególnych składowych koloru podanymi wartościami (w konstruktorze tym zgłaszaj wyjątki, gdy któraś z podanych wartości wykroczy poza zakres 0...255). Klasa ma też posiadać **gettera i settera** odpowiednio do odczytania i zapisania określonej barwy składowej a także metody do **rozjaśniania i przyciemniania** koloru. Dodatkowo w klasie umieść **metodę statyczną** do sumowania kolorów (średnia arytmetyczna poszczególnych barw).

Dalej zdefiniuj klasę **kolortransparentny**, która uzupełni klasę kolor o współczynnik alfa (wartość z zakresu 0...255), określającą przezroczystość (0 oznacza całkowitą przezroczystość a 255 pełny kolor).

Kolejna klasa `kolornazwany` ma rozszerzać klasę `kolortransparentny` o możliwość ustalenia nazwy mnemonicicznej dla koloru (łańcuch znakowy typu `string`). Domyślnie nazwa ma być pusta, ale jeśli się pojawi to ma się składać tylko z liter.

Następnie zdefiniuj klasę `punkt`, reprezentującą punkt na płaszczyźnie w postaci współrzędnych `x` i `y` typu `double`. Domyślne współrzędne punktu to (0, 0). Dodatkowo w klasie umieść składową funkcję instancyjną `odleglosc()` do obliczenia odległości do innego wskazanego punktu oraz składową funkcję statyczną `wspol liniowe()` do sprawdzenia czy podane trzy punkty są współliniowe.

Kolejna klasa `punktnazwany` ma rozszerzać klasę `punkt` o możliwość ustalenia nazwy mnemonicicznej dla punktu (łańcuch znakowy typu `string`). Domyślnie nazwa ma być pusta, ale jeśli się pojawi to ma się składać tylko z liter lub cyfr i ma się rozpoczynać od litery.

Inna klasa `punktkolorowy` ma rozszerzać klasę `punkt` o kolor (właściwość koloru określić za pomocą klasy `kolortransparentny`).

Wreszcie klasa `punktkolorowynazwany` ma dziedziczyć publicznie zarówno po klasie `punktkolorowy` jak również po klasie `punktnazwany`. Dziedziczenie to zorganizuj w taki sposób, aby współrzędne takiego punktu nie dublowały się ze względu na wielodziedziczenie.

Dalej klasa `punkt2d`, rozszerzająca klasę `punkt` o operację transpozycji, czyli przesunięcia o zadany wektor w przestrzeni 2D (klasa `wektor2d`).

Ostatnia klasa to `punkt3d`, rozszerzająca klasę `punkt2d` o kolejną współrzędną z typu `double` (poprzednie współrzędne `x` i `y` mają być w tej klasie bezpośrednio dostępne). W klasie tej zdefiniuj operację transpozycji o zadany wektor w przestrzeni 3D (klasa `wektor3d`). Przesłoń składową funkcję instancyjną `odleglosc()`, dostosowując ją do uwzględnienia następnego wymiaru w obliczaniu odległości między punktami. Przesłoń także składową funkcję statyczną `wspol liniowe()` do sprawdzenia czy podane trzy punkty w przestrzeni są współliniowe.

Na koniec napisz program, który bardzo rzetelnie przetestuje całą funkcjonalność zaprogramowaną we wszystkich klasach opisujących kolory i punkty na płaszczyźnie i w przestrzeni.

### Uwaga

Podziel program na pliki nagłówkowe (definicje klas) i źródłowe (definicje metod zadeklarowanych w klasach). Funkcję `main()` z testami umieść w osobnym pliku źródłowym.

### Ważne elementy programu

- Implementacja dziedziczenia.
- Udostępnianie stanu w klasach potomnych.
- Dziedziczenie wielobazowe z wykorzystaniem dziedziczenia wirtualnego.
- Przetestowanie wszystkich klas w funkcji `main()`.