

kurs języka C++

kolorowe piksele

Instytut Informatyki
Uniwersytetu Wrocławskiego

Paweł Rzechonek

Prolog

RGB to jeden z modeli przestrzeni barw, opisywanej współrzędnymi RGB. Jego nazwa powstała ze złożenia pierwszych liter angielskich nazw trzech barw: R – red (czerwonej), G – green (zielonej) i B – blue (niebieskiej). Jest to model wynikający z właściwości sensorycznych ludzkiego oka, w którym wrażenie widzenia dowolnego koloru można wywołać przez zmieszanie w ustalonych proporcjach trzech wiązek światła o barwie czerwonej, zielonej i niebieskiej.

Zadanie

Zdefiniuj klasę `kolor` reprezentującą kolor zapisany w systemie RGB. W klasie tej umieść trzy pola typu `unsigned short`, które będą pamiętały nasycenie daną barwą (wartości z zakresu 0...255). Zdefiniuj dwa konstruktory: pierwszy 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 wykróczy poza zakres 0...255). Klasa ma też posiadać gettery i settery odpowiednio do odczytania i zapisania określonej barwy składowej, a także metody do rozjaśniania i przyciemniania całego koloru. Dodatkowo w klasie umieść metodę statyczną do łączenia kolorów (średnia arytmetyczna poszczególnych barw).

Dalej zdefiniuj klasę `kolortransparentny`, która uzupełni klasę `kolor` o współczynnik alfa typu `unsigned short` (wartość z zakresu 0...255), określający przezroczystość (0 oznacza całkowitą przezroczystość a 255 pełny kolor). Zastosuj w tym przypadku dziedziczenie niepubliczne.

Następnie zdefiniuj klasę `kolornazwany`, która dołoży do klasy `kolor` nazwę mnemoniczną (łańcuch znakowy typu `string`) – domyślnie nazwa ma być pusta, ale jeśli już zostanie podana, to ma się składać tylko z małych liter alfabetu angielskiego. Zastosuj dziedziczenie publiczne.

Klasy `kolortransparentny` oraz `kolornazwany` mają dziedziczyć publicznie po klasie `kolor`. Kolejna definiowana klasa `kolorn` ma posiadać zarówno współczynnik alfa jak również nazwę mnemoniczną. Zastosuj tutaj publiczne dziedziczenie wielobazowe po klasach `kolortransparentny` i `kolornazwany`; zadbaj o to, by w tej klasie nie powtarzały się

informacje dotyczące barw składowych koloru, pochodzące z klasy `kolor` (użyj w odpowiednim momencie dziedziczenia wirtualnego).

Na koniec zdefiniuj klasę `piksel`, reprezentującą pojedynczy piksel na ekranie monitora czy wyświetlacza; piksel ma być zdefiniowany na ekranie za pomocą współrzędnych x i y typu `int`. Przy inicjalizacji w konstruktorze i przekształceniach w setterach sprawdzaj, czy współrzędne piksela nie wychodzą poza ekran (współrzędne ekranu to nieujemne wartości mniejsze od rozmiaru ekranu). Rozmiar ekranu określ za pomocą pól statycznych w klasie `piksel` (ustal na stałe pewien arbitralnie wybrany standard rozdzielczości, na przykład SVGA, XGA, SXGA+ dla ekranów o proporcjach 4:3, albo HDTV720, HDTV1080 dla ekranów o proporcjach 16:9). Dodatkowo w klasie umieść składowe funkcje instancyjne do wyznaczenia odległości piksela od lewego, prawego, górnego i dolnego brzegu ekranu oraz składową funkcję statyczną do obliczania odległości między parą pikseli (napisz dwie przeciążone wersje tej funkcji – dla wskaźników i dla referencji).

Ostatnia klasa to `pikselkolorowy`, która ma dziedziczyć publicznie po klasie `piksel`. Klasę tą należy wyposażyć w możliwość ustalenia koloru piksela – kolor ten określ za pomocą pola składowego typu `kolor`. Dodaj w tej klasie funkcjonalność związaną z przemieszczaniem pikseli o zadany wektor (gdyby nowe współrzędne miały wyjść poza ekran, to należy zgłosić wyjątek).

Napisz także program, który rzetelnie przetestuje wszystkie zdefiniowane klasy. Pomocne przy testowaniu klas mogą okazać się operatory strumieniowe.

Ważne elementy programu

- Podział programu na pliki nagłówkowe i źródłowe oraz osobny plik źródłowy z funkcją `main()` do przeprowadzenia testów.
- Implementacja dziedziczenia (oprócz zwykłego dziedziczenia publicznego i niepublicznego należy użyć dziedziczenia wielobazowego i wirtualnego).
- W funkcjach składowych i w konstruktorach zgłaszaj błędy za pomocą wyjątków i instrukcji `throw`.
- Wykorzystanie automatycznego rzutowania wskaźników/referencji do klas pochodnych na wskaźniki/referencje do klas bazowych w wywołaniu funkcji liczącej odległość między pikselami.
- W funkcji `main()` należy rzetelnie przetestować wszystkie zdefiniowane klasy.