

Proszę napisać klasę **Image** implementującą dwuwymiarowy obraz monochromatyczny (tzn. obraz w odcieniach szarości).

Wskazówki:

- 1) *Obrazy monochromatyczne przechowywane są w pamięci komputerów jako dwuwymiarowa tablica, której elementami są jasności kolejnych pikseli obrazu. Jasność każdego piksela może być reprezentowana przez 8 bitową liczbę naturalną, tzn. liczbę z przedziału 0 – 255 (unsigned char). Liczba 0 oznacza kolor czarny, 255 - biały.*
- 2) *W programie wygodnie jest zdefiniować własną, krótszą nazwę (np. byte) dla typu unsigned char.*
- 3) *Dwuwymiarowa tablica ma być alokowana dynamicznie zgodnie z metodą poznaną w poprzednim semestrze na przedmiocie „Programowanie Mikrokontrolerów”.*
- 4) *Zauważ że konstruktor, konstruktor kopiujący oraz operator = mają pewne wspólne instrukcje np. alokujące pamięć. Programując należy unikać duplikacji kodu, a zatem te wspólne instrukcje powinny być napisane w jednym miejscu np. w osobnej funkcji initialize.*

Klasa Image powinna mieć następujące funkcje składowe:

Na ocenę 3:

- 1) Konstruktor:
`Image(int width, int height);`
- 2) Metodę nadającą tą samą jasność wszystkim pikselom w obrazie, np.:
`void fillInImage(unsigned char value)`
- 3) Metodę służącą do odczytu piksela
- 4) Metodę służącą do zapisu piksela
- 5) Metody:
`int getWidth();`
`int getHeight();`
- 6) Metody dokonujące lustrzanych odbić obrazu:
`void flipHorizontal();`
`void flipVertical();`
- 7) Program powinien być napisany wielomodułowo (np. w plikach Image.h, Image.cpp oraz main.cpp)

- 8) Należy tak zaprojektować klasę Image aby działała poprawnie również w sytuacji gdy użytkownik posługuje się pustymi obrazami, tzn. obrazami o wymiarach width = 0, height = 0, jak w poniższej sytuacji:

```
Image image(0, 0);  
Image copiedImage(image);  
Image equaledImage(5, 4);  
EqualedImage = image;
```

Na ocenę 4:

9) Destruktor

10) Konstruktor kopiujący

11) Operator =

Na ocenę 5:

- 12) Operator () służący do operacji na poszczególnych pikselach. Przykładowe użycie operatora w programie:

```
Image image(10, 15);  
image.fillInImage(0);  
image(2, 3) = 100;  
int pixelIntensity = image(2, 3);
```

- 13) Operator << wyświetlający obraz na ekranie (wskazówka: nie będzie to funkcja składowa, lecz globalna. Czy będzie to funkcja zaprzyjaźniona?)

Uwaga!

Alokacja pamięci uproszczonym sposobem (bez przechowywania wskaźników do kolejnych wierszy) powoduje obniżenie oceny o 1.