Practical work report #1

Supervisor:

Dr. Anass Belcaid

[23-10-2021]

EIDIA

Authored by: Yahia BENCHEDDI

UEMF
الجامعة الأورومتوسطية بفاس
EUROMED UNIVERSITY OF FES
UNIVERSITÉ EUROMED DE FÈS

Executive Summary

This project was a part of the first week assignment of a course on Human-Machine interface in C++ taught by Dr.Belcaid from department of computer science at UEMF. The goal of this project was to transform the given image by changing the Hue, Saturation or Luminance of the pixel.

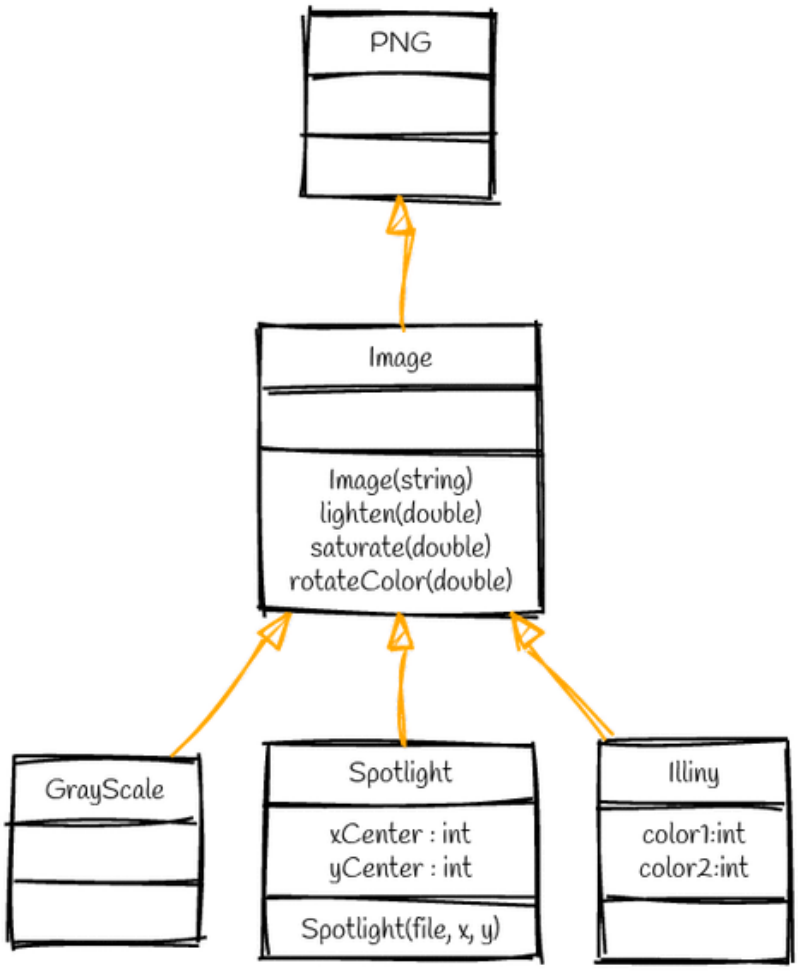
Program overview

Objectives

The goal of the program was to complete a given zip file containing some pre-coded classes by adding and writing the code to a GrayScale, SpotLight and Illiny classes that inherit form an Image class according to the UML class diagram given:

Inhertance diagram

Your goal is to write additional classes that **inherit** from this class and implement addtional functionalities.



UML class diagram for the additional Images classes.

All the provided tests are passed :

SimpleTest imagemanip

Correct (PROVIDED_TEST, line 98) Image saturate() saturates a pixels by 0.1

Correct (PROVIDED_TEST, line 107) Image rotateColor(double) rotates the color

Correct (PROVIDED_TEST, line 115) Image rotateColor(double) keeps the hue in the range [0, 360]

Correct (PROVIDED_TEST, line 127) Grayscale Image

Correct (PROVIDED_TEST, line 138) illini

Correct (PROVIDED_TEST, line 151) Pixels closest to blue become blue

Correct (PROVIDED_TEST, line 161) Pixels closest to orange become orange

Correct (PROVIDED_TEST, line 170) Hue wrap-arounds are correct (remember: h=359 is closer to orange than blue)

Correct (PROVIDED_TEST, line 179) Spotlight does not modify the center pixel

Correct (PROVIDED_TEST, line 186) Spotlight creates an 80% dark pixel >160 pixels away

Correct (PROVIDED_TEST, line 192) Spotlight is correct at 20 pixels away from center

Correct (PROVIDED_TEST, line 199) Spotlight is correct at 5 pixels away from center

Passed 32 of 32 tests. No complaints here!

and for a visual example of each class and method I'll be using an image of this cool cat for it:

Classes & headers:

-Just like the diagram says we have an Image class that inherit from PNG and contains a constructor and three methods:

Image.h:

```
#ifndef IMAGE_H
#define IMAGE_H
#include "PNG.h"

class Image : public PNG
{
public:
    using PNG::PNG;
    Image (string path);
    void lighten(double amount=0.1);
    void saturate(double amounnt=0.1);
    void rotateColor( double angle);
};

#endif // IMAGE_H
```

Image.cpp:

```
#include "image.h"
Image::Image(string filename):PNG()

{
    readFromFile(filename);

}
void Image::lighten(double amount)
{
    for(unsigned i=0;i< width();i++)
    {
        for(unsigned j=0;j<height();j++)
```

```
            {
                HSLAPixel &p  = getPixel(i,j);
                p.l += amount;
                p.l = (p.l>0) ? p.l:0;
                p.l = (p.l<=1)? p.l:1;
            }
        }
    }
    void Image::saturate(double amount)
    {
        for(unsigned i=0;i< width();i++)
        {
            for(unsigned j=0;j<height();j++)
            {
                HSLAPixel &p  = getPixel(i,j);
                p.s += amount;
                p.s = (p.s>0) ? p.s:0;
                p.s = (p.s<=1)? p.s:1;
            }
        }
    }
    void Image::rotateColor(double angle)
    {
        for(unsigned i=0;i< width();i++)
        {
            for(unsigned j=0;j<height();j++)
            {
                HSLAPixel &p  = getPixel(i,j);
                p.h += angle;
                while (p.h>360)
                    p.h=p.h-360;
                while (p.h<0)
                    p.h=p.h+360;

            }
        }
    }
```

As you can see in the code above we declared in the headers file and coded in the .cpp file the Image constructor and the lighten, saturate and rotateColor methods.

Example of lighten:

Example of saturate:

Example of rotateColor:

-Then we have the Illini class with two attributes color1 and color2:

illini.h:

```
#ifndef ILLINI_H
#define ILLINI_H
#include "image.h"

class Illini : public Image
{
public:
    using Image ::Image;
    Illini(string filename,int color1=11,int
color2=216);

};

#endif // ILLINI_H
```

illini.cpp:

```
#include "illini.h"
Illini::Illini(string filename,int color1,int color2):Image()
{
    readFromFile(filename);
    int a,b;
```

```
for(unsigned i=0;i<width();i++)
    for(unsigned j=0;j<height();j++)
    {
        HSLAPixel &P=getPixel(i,j);

        if(P.h>color1){
            if(P.h-color1<360+color1-P.h)
                a=P.h-color1;
            else
                a=360+color1-P.h;
        }else{
            if(-P.h+color1<360-color1+P.h)
                a=-P.h+color1;
            else
                a=360-color1+P.h;
        }
        if(P.h>color2){
            if(P.h-color2<360+color2-P.h)
                b=P.h-color2;
            else
                b=360+color2-P.h;
        }else{
            if(-P.h+color2<360-color2+P.h)
                b=-P.h+color2;
            else
                b=360-color2+P.h;
        }
        if(a<b)
            P.h=color1;
        else
            P.h=color2;}}
```

Example of illini:

-We have next the Grayscale class:

grayscale.h:

```
#ifndef GRAYSCALE_H
#define GRAYSCALE_H
#include"image.h"

class Grayscale : public Image

{
public:
    using Image ::Image;
   Grayscale (string path);
};

#endif // GRAYSCALE_H
```

grayscale.cpp:

```
#include "grayscale.h"
Grayscale::Grayscale(string filename):Image()
{
    readFromFile(filename);
    for(unsigned i=0;i< width();i++)
    {
        for(unsigned j=0;j<height();j++)
        {
            HSLAPixel &p  = getPixel(i,j);
            p.s =0;
        }
    }
}
```

Example of grayscale:

-And last but not least we have the Spotlight class with two attributes xCenter and yCenter as integers and a constructor with three paramters:

spotlight.h:

```
#ifndef SPOTLIGHT_H
#define SPOTLIGHT_H
#include "image.h"

class Spotlight : public Image
{
public:
    using Image::Image;
    Spotlight(string filename,int x,int y);

};

#endif // SPOTLIGHT_H
```

spotlight.cpp:

```
#include "spotlight.h"
#include "math.h"

Spotlight::Spotlight(string filename, int x, int y) :
Image()
{
    readFromFile(filename);
    for(unsigned i=0; i<width(); i++  ){
        for(unsigned j=0; j<height(); j++){
            HSLAPixel &p = getPixel(i,j);
            auto dist = sqrt((i-x)*(i-x) + (j-y)*(j-y));
            double luminanceval = 1-dist*0.5/100 ;
            if (dist>160){
                luminanceval=0.2;
            }
            p.l=p.l*luminanceval;
        }
    }
}
```

Example of spotlight:

Conclusion

The practical was a bit challenging but fun at the same time, I admit I used some help from my friend special thanks to him and thank you professor hope you like the work.