Assignment 1: Cartoonify

Given a headshot photo of an individual (e.g. the headshot of Ed Sheeran), generate the cartoon version of the photo by sketching it using Bézier curves.

The skeleton code is provided below.

I have provided you with a simple Vertex class that allows you to specify the x and y values of a point. You will utilize this class for modeling the control points of your sketch.

**Note: the C++ vector class is the equivalent to a list in most other languages.

You will complete the following functions for the assignment:

1) `generate_points` : a function that generates the set of control points for a Bézier curve
   parameters: vector<Vertex>
   returns: vector<Vertex>
2) `draw_curve` : calls `generate_points` to generate the set of points for the Bézier curve algorithm and connects the points with lines
   parameters: vector<Vertex>, int
   returns: none

Submission:
You will submit the following to Bright Space

1) "assignment1.cpp"
2) Your sketch in JPEG, JPG, or PNG: results.{jpeg, jpg, png}
3) The photo your sketch was based on in JPEG, JPG or PNG: photo.{jpeg, jpg, png}

Grading:
I will be compiling the assignment using the following command:

```
gcc -o assignment1 assignment1.cpp -lGL -lGLU -lglut
```

Your code must compile for me to assign points!

Your assignment will be graded on:

1) the correctness of your implementation of Bézier's algorithm
2) effort placed recreating the subject via your sketch
   e.g. a simple happy face does not do Ed Sheeran justice

Late Policy:
For each day the assignment is late, 50% of its worth will be deducted, e.g. 100% on time, 50% 1 day late, 25% 2 days late, etc.

assignment1.cpp

```
/***
 assignment1.cpp
 Assignment-1: Cartoonify

 Name: Wong, Alex (Please write your name in Last Name, First Name format)

 Collaborators: Doe, John; Doe, Jane
 ** Note: although the assignment should be completed individually
 you may speak with classmates on high level algorithmic concepts. Please
 list their names in this section

 Project Summary: A short paragraph (3-4 sentences) describing the work you
 did for the project: e.g. did you use the iterative or recursive approach?
 ***/


#ifdef __APPLE__
#include <OpenGL/gl.h>
#include <OpenGL/glu.h>
#include <GLUT/glut.h>
#else
#include <GL/gl.h>
#include <GL/glu.h>
#include <GL/glut.h>
#endif

#include <vector>
#include <iostream>
using namespace std;

class Vertex {
    GLfloat x, y;
    public:
    Vertex(GLfloat, GLfloat);
    GLfloat get_y() { return y; };
    GLfloat get_x() { return x; };
};

Vertex::Vertex(GLfloat X, GLfloat Y) {
    x = X;
    y = Y;
}
```

```
void setup() {
    glClearColor(1.0f, 1.0f, 1.0f, 1.0f);
}

vector<Vertex> generate_points(vector<Vertex> control_points) {
    vector<Vertex> points;

    // Generate points for a given Bezier curve iteration

    return points;
}

void draw_curve(vector<Vertex> control_points, int n_iter) {
    // Draw a Bezier curve based on the given control points
}

void display() {
    glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);
    // Set our color to black (R, G, B)
    glColor3f(0.0f, 0.0f, 0.0f);

    // Draw cartoon

    glutSwapBuffers();
}

int main(int argc, char *argv[]) {
    glutInit(&argc, argv);
    glutInitDisplayMode(GLUT_RGB | GLUT_DEPTH | GLUT_DOUBLE);
    glutInitWindowSize(800,600); // Set your own window size
    glutCreateWindow("Assignment 1");
    setup();
    glutDisplayFunc(display);
    glutMainLoop();
    return 0;
}
```