Practical Homework Assignment 1: Development Environment Setup

Objective: Set up a complete development environment for Computer Graphics programming

Tasks:

1. Install and configure required software
2. Create and run a test program
3. Document the setup process

Detailed Requirements:

1. **Software Installation**
    - Install Visual Studio Community Edition
    - Set up OpenGL and GLUT libraries
    - Configure project settings
2. **Test Program Creation**

```
#include <GL/glut.h>

void display() {
    glClear(GL_COLOR_BUFFER_BIT);
    glColor3f(1.0, 0.0, 0.0);  // Red color

    // Draw a simple triangle
    glBegin(GL_TRIANGLES);
        glVertex2f(-0.5, -0.5);
        glVertex2f(0.5, -0.5);
        glVertex2f(0.0, 0.5);
    glEnd();

    glFlush();
}

int main(int argc, char** argv) {
    glutInit(&argc, argv);
    glutCreateWindow("First OpenGL Program");
    glutDisplayFunc(display);
    glutMainLoop();
    return 0;
}
```

3. **Documentation Requirements**
    - Screenshot of successful installation
    - Screenshot of test program running
    - List of any errors encountered and how they were solved

Submission Format:

- PDF document containing:
    - Step-by-step installation process
    - Screenshots
    - Test program code
    - Troubleshooting notes

Practical Homework Assignment 2: Basic Graphics Programming

Objective: Create a simple interactive graphics program

Tasks:

1. Create a program that draws basic shapes
2. Implement mouse interaction
3. Add color changes

Program Requirements:

1. **Basic Shapes Implementation**

```
#include <GL/glut.h>

// Window dimensions
const int WINDOW_WIDTH = 800;
const int WINDOW_HEIGHT = 600;

void drawRectangle() {
    glBegin(GL_POLYGON);
        glVertex2f(-0.5, -0.5);
        glVertex2f(0.5, -0.5);
        glVertex2f(0.5, 0.5);
        glVertex2f(-0.5, 0.5);
    glEnd();
}

void drawCircle(float radius) {
    glBegin(GL_POLYGON);
    for(int i = 0; i < 360; i++) {
        float angle = i * 3.14159 / 180;
        glVertex2f(cos(angle) * radius, sin(angle) * radius);
    }
    glEnd();
}

void display() {
    glClear(GL_COLOR_BUFFER_BIT);
```

```
    // Draw shapes with different colors
    glColor3f(1.0, 0.0, 0.0);  // Red
    drawRectangle();

    glColor3f(0.0, 0.0, 1.0);  // Blue
    drawCircle(0.3);

    glFlush();
}

void mouse(int button, int state, int x, int y) {
    // Add mouse interaction code here
}

int main(int argc, char** argv) {
    glutInit(&argc, argv);
    glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB);
    glutInitWindowSize(WINDOW_WIDTH, WINDOW_HEIGHT);
    glutCreateWindow("Basic Shapes");

    glutDisplayFunc(display);
    glutMouseFunc(mouse);
    glutMainLoop();
    return 0;
}
```

2. **Required Features:**
    - Draw at least 3 different shapes
    - Implement color changes
    - Add mouse click interaction
    - Include simple animation

Submission Requirements:

1. **Source Code**
    - Well-commented code
    - Proper indentation
    - Clear function names
2. **Documentation**
    - Program description
    - Instructions to run
    - Screenshots of output
    - Explanation of key functions