

**Department of Computer Science and Engineering**

**21st Batch**

**Lab Report 4**

|  |  |
| --- | --- |
| Course title | : Computer Graphics Lab |
| Course Code | : CSE - 422 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Submitted By** | | **Submitted To** | |
|  |  |  |  |
| Name | : Md. Mahfujur Rahman | Name | : Ayesha Akter Lima |
| ID | : 192311014 | Designation | : Lecturer,  Varendra University,  Rajshahi |
| Section | : A |  |
| Semester | : 11th |  |
| Batch | : 21st | Name | : Monika Kabir |
|  |  | Designation | : Assistant Professor  Varendra University,  Rajshahi |
|  |  |  |

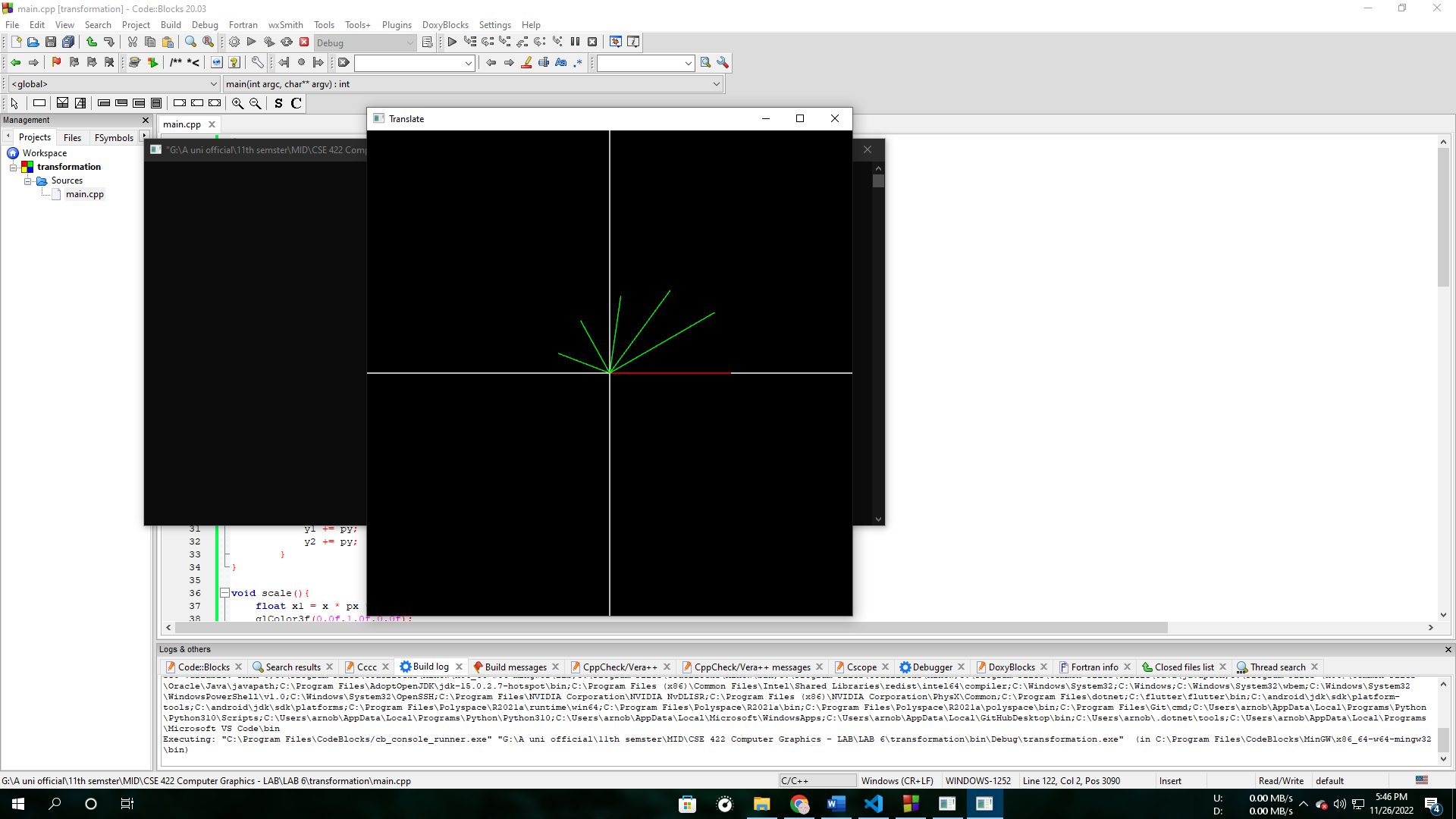
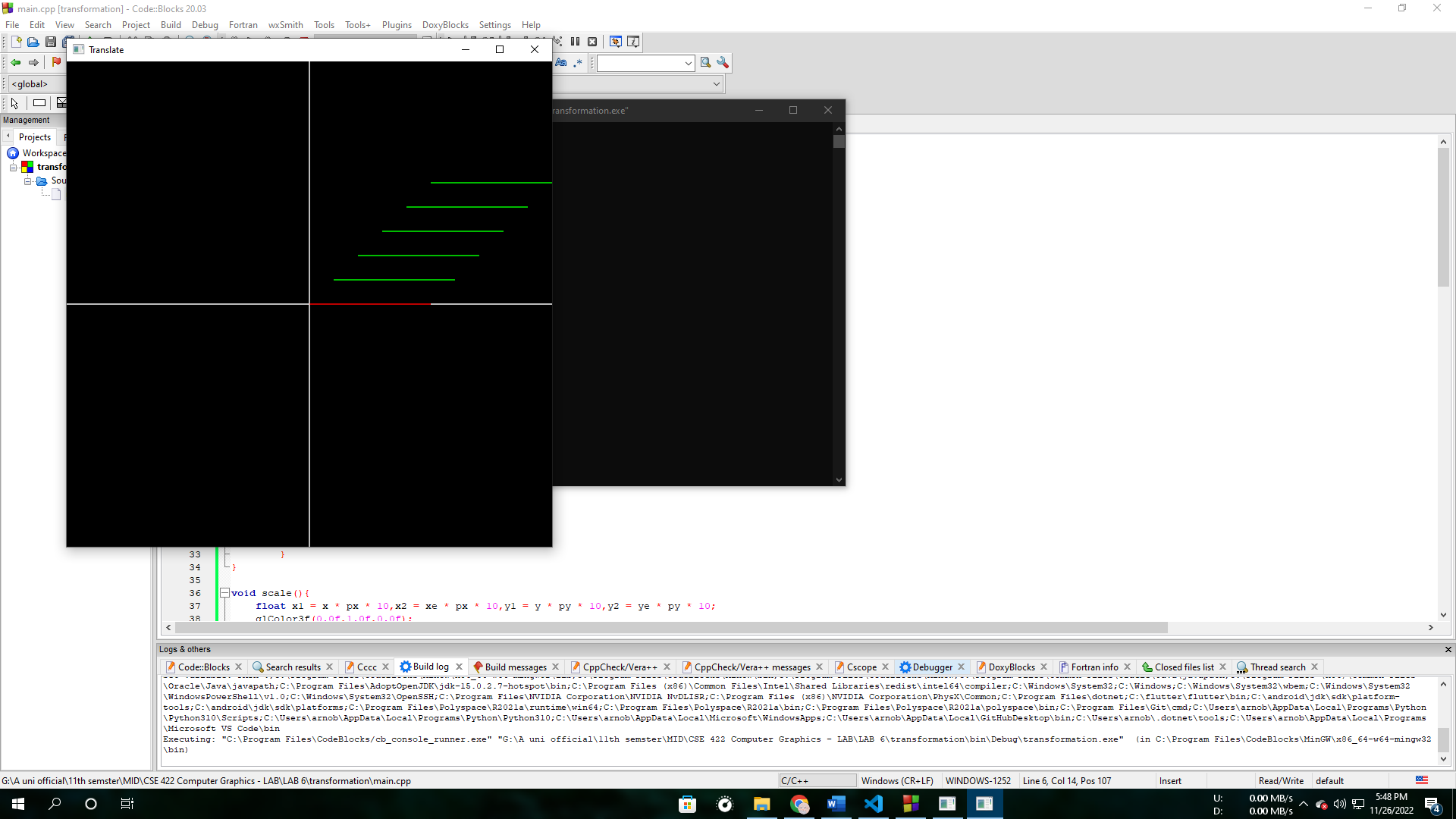
------------

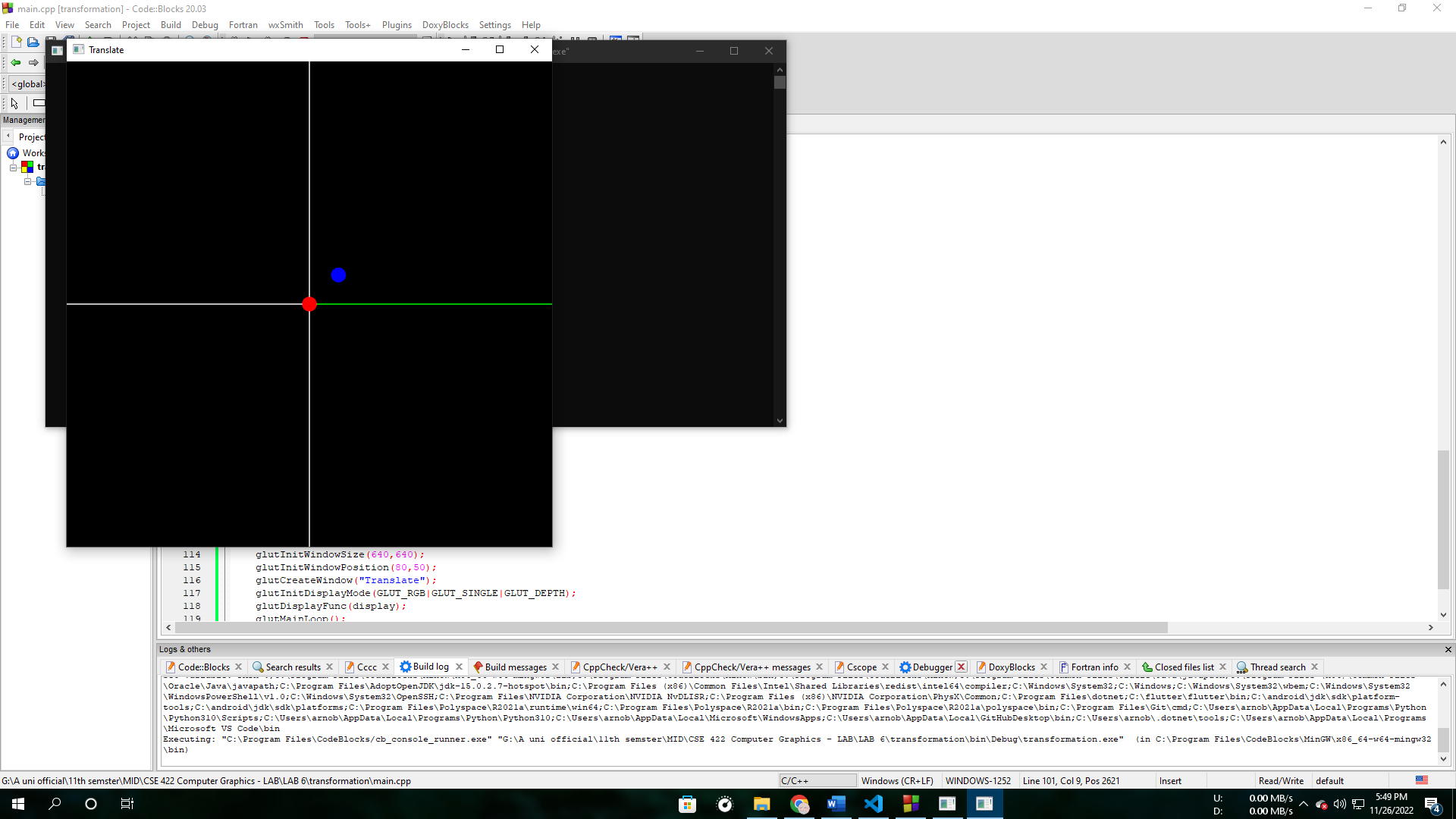
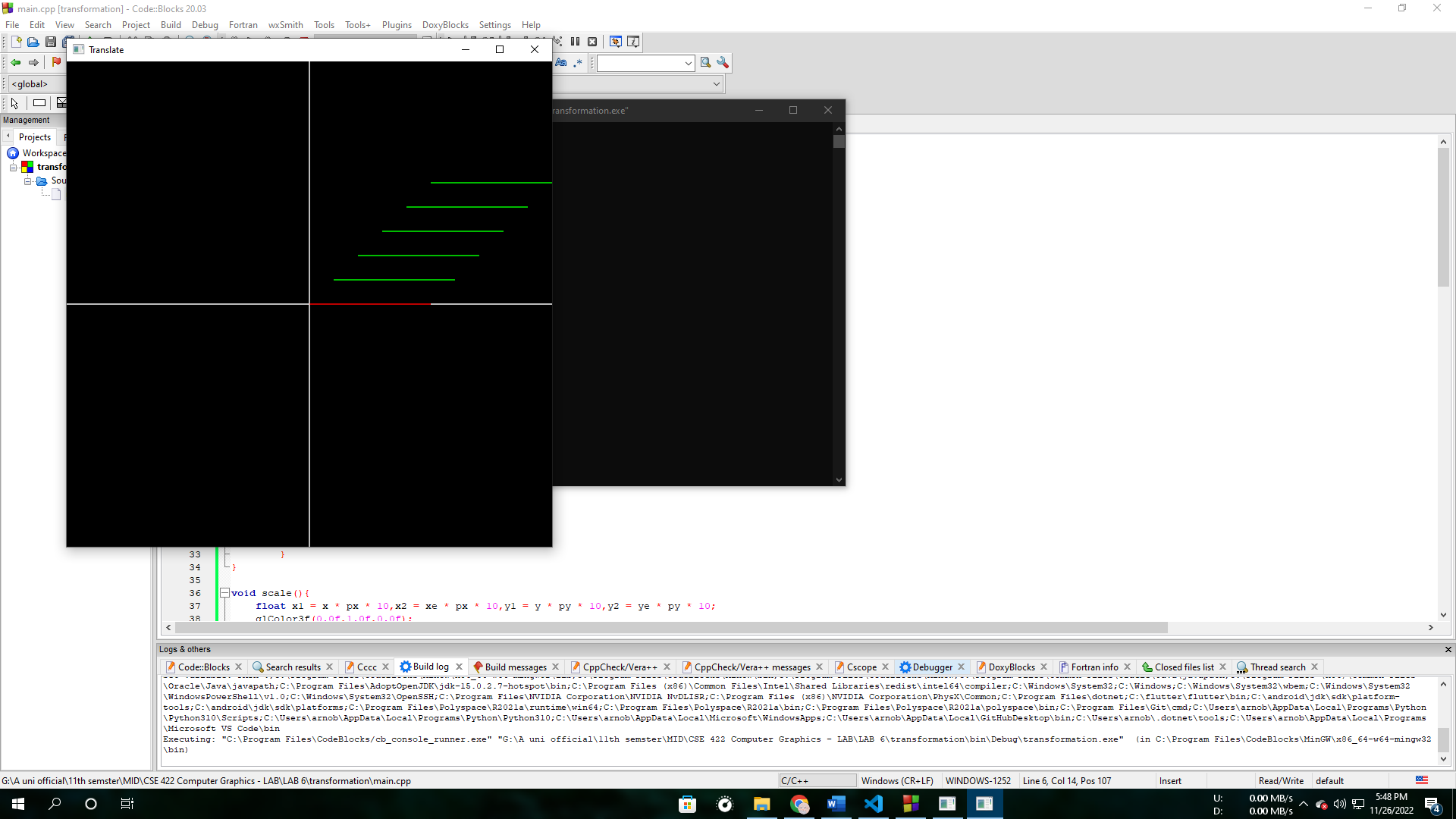
|  |  |
| --- | --- |
| Signature | Submission date: |
|  |  |

**Problem Statement:** Transformation of Object.

**Code:**

|  |
| --- |
| #include <windows.h>  #include <GL/glut.h>  #include <bits/stdc++.h>  using namespace std;  float px,py, theta = 30;  float x, y ,xe, ye;  void translatep(){          float x1 = x + px;          float y1 = y + py;          glColor3f(1.0f,0.0f,0.0f);          glVertex2f(x,y);          glColor3f(0.0f,0.0f,1.0f);          glVertex2f(x1,y1);  }  void translate(){          float x1 = x + px ,x2 = xe + px,y1 = y + py ,y2 = ye +py;          for(int i = 1 ; i <= 5 ; i++){              glColor3f(0.0f,1.0f,0.0f);              glVertex2f(x1,y1);              glVertex2f(x2,y2);              x1 += px;              x2 += px;              y1 += py;              y2 += py;          }  }  void scale(){      float x1 = x \* px \* 10,x2 = xe \* px \* 10,y1 = y \* py \* 10,y2 = ye \* py \* 10;      glColor3f(0.0f,1.0f,0.0f);      glVertex2f(x1,y1);      glVertex2f(x2,y2);  }  void scalel(){      float x1 = x \* px \* 10,x2 = xe \* px \* 10,y1 = y \* py \* 10,y2 = ye \* py \* 10;      glColor3f(0.0f,1.0f,0.0f);      for(int i = 1 ; i <= 5 ; i++){              glColor3f(0.0f,1.0f,0.0f);              glVertex2f(x1,y1);              glVertex2f(x2,y2);              x1 \*= px \* 10;              x2 \*= px \* 10;              y1 \*= py \* 10;              y2 \*= py \* 10;          }  }  void rotation(){          float x1 = x \* cos(theta \* M\_PI/180) - y \* sin(theta \* M\_PI/180);          float x2 = xe \* cos(theta \* M\_PI/180) - ye \* sin(theta \* M\_PI/180);          float y1 = x \* sin(theta \* M\_PI/180) + y \* cos(theta \* M\_PI/180);          float y2 = xe \* sin(theta \* M\_PI/180) + ye \* cos(theta \* M\_PI/180);      for(int i = 0; i < 5 ; i++){          glColor3f(0.0f,1.0f,0.0f);          glVertex2f(x1,y1);          glVertex2f(x2,y2);          x1 = x1 \* cos(theta \* M\_PI/180) - y1 \* sin(theta \* M\_PI/180);          x2 = x2 \* cos(theta \* M\_PI/180) - y2 \* sin(theta \* M\_PI/180);          y1 = x1 \* sin(theta \* M\_PI/180) + y1 \* cos(theta \* M\_PI/180);          y2 = x2 \* sin(theta \* M\_PI/180) + y2 \* cos(theta \* M\_PI/180);      }  }  void display(){      glClear(GL\_COLOR\_BUFFER\_BIT);      glBegin(GL\_LINES);          glColor3f(1.0f,1.0f,1.0f);          glVertex2f(1.0f,0.0f);          glVertex2f(-1.0f,0.0f);          glVertex2f(0.0f,1.0f);          glVertex2f(0.0f,-1.0f);      glEnd();      glBegin(GL\_LINES);          x=0.0, y=0.0 ,xe = 0.5, ye = 0.0;          glColor3f(1.0f,0.0f,0.0f);          glVertex2f(x,y);          glVertex2f(xe,ye);          translate();          scalel();          rotation();      glEnd();      glPointSize(20.0f);      glBegin(GL\_POINTS);          x=0.0, y=0.0;          translatep();      glEnd();      glFlush();  }  int main(int argc,char\*\* argv){      cout<<"Input the translate Of X:"<<endl;      cin>>px;      cout<<"Input the translate Of Y:"<<endl;      cin>>py;      glutInit(&argc, argv);      glutInitWindowSize(640,640);      glutInitWindowPosition(80,50);      glutCreateWindow("Translate");      glutInitDisplayMode(GLUT\_RGB|GLUT\_SINGLE|GLUT\_DEPTH);      glutDisplayFunc(display);      glutMainLoop();      return 0;  } |

**Results: **

****