**NAMA : FERDHY DHETYA YAHYA**

**NIM : 182410103071**

**KELAS : GRAFIKA KOMPUTER D**

**Hello World**

#include <GL/glut.h>

Int main(int argc, char\*\* argv){

glutInit(&argc, argv);

glutInitWindowSize(500,500);

glutInitWindowPosition(500,500);

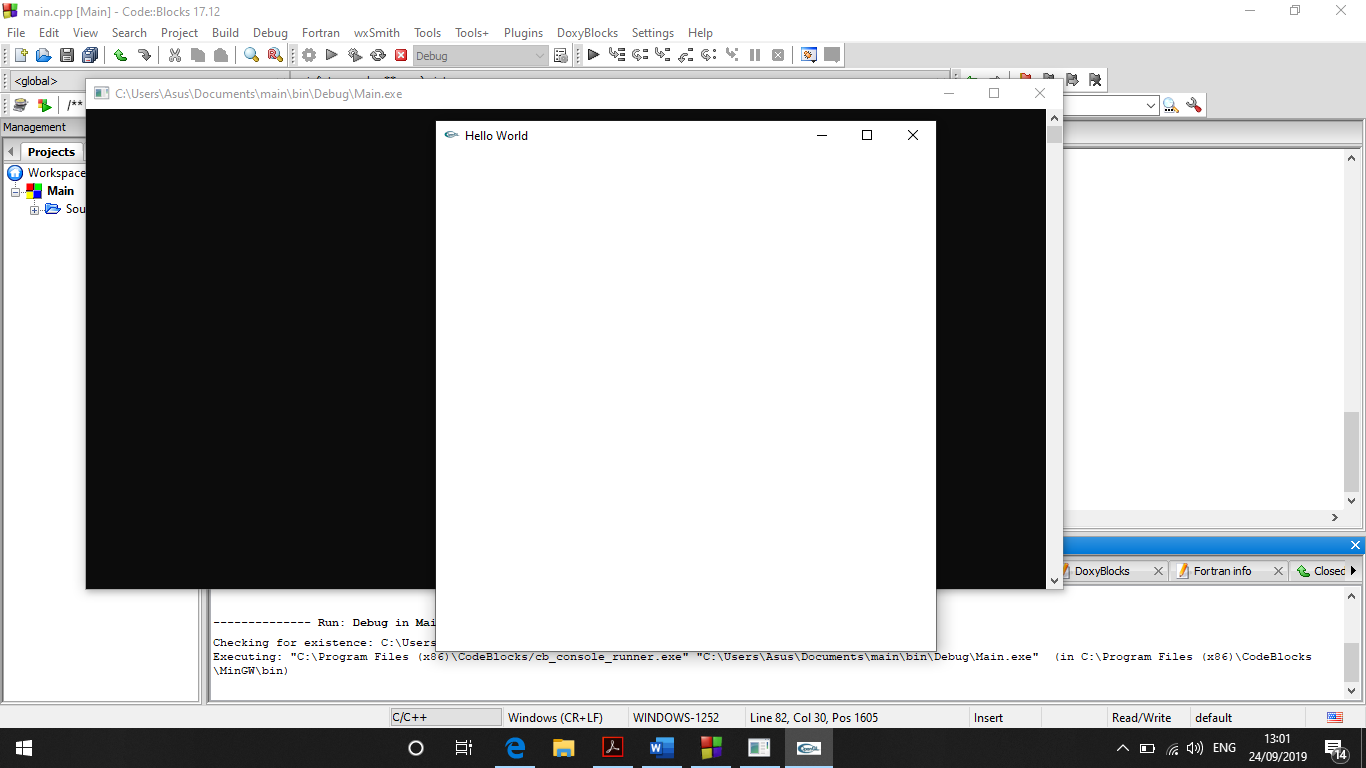
glutCreateWindow(“Hello World”);

glutMainLoop();

}

Hasil:

* #include <GL/glut.h>: Direktif include dari glut.h yang fungsinya menambahkan compiler tersebut pada library code
* Int main(int argc, char\*\* argv){: argc (argument count) dan argv (argument vector), adalah command-line arguments
* glutInit(&argc, argv);:
* glutInitWindowSize(500,500);: inisiasi ukuran window dengan format (lebar, tinggi) satuan pixels
* glutInitWindowPosition(500,500);: inisiasi posisi window dengan format (x, y) satuan pixels
* glutCreateWindow(“Hello World”);: membuat window top-level dengan nama “Hello World” dengan label nama yang akan ditampilkan pada window
* glutMainLoop();}: proses looping pada GLUT untuk menampilkan



**Points**

#include <GL/glut.h>

void display(){

glClear(GL\_COLOR\_BUFFER\_BIT);

glMatrixMode(GL\_MODELVIEW);

glPointSize(10);

glColor3f(1,0,0);

glBegin(GL\_POINTS);

glVertex2f(1,1);

glEnd();

}

int main(int argc, char\*\* argv)

{

glutInit(&argc, argv);

glutInitWindowSize(500, 500);

glutInitWindowPosition(500, 500);

glutCreateWindow("Hello World");

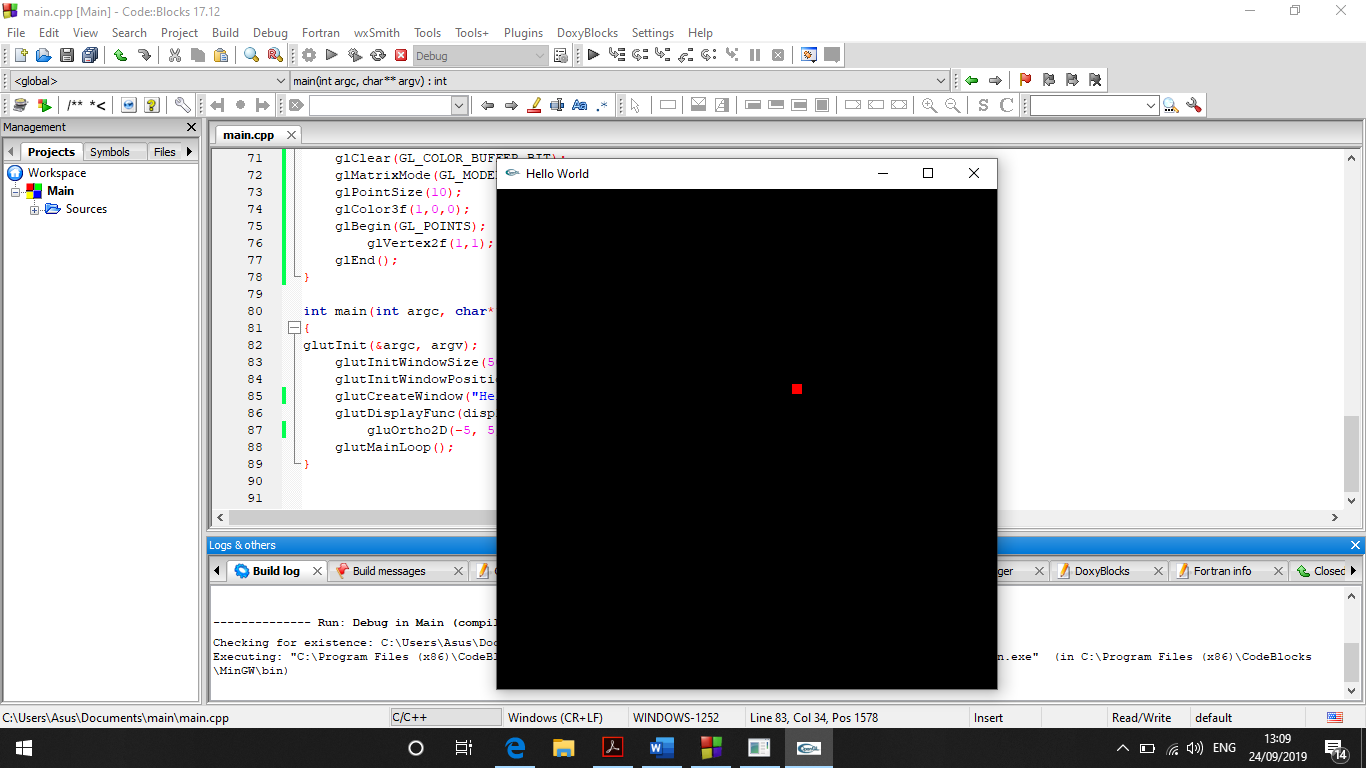
gluOrtho2D(-5, 5, -5, 5);

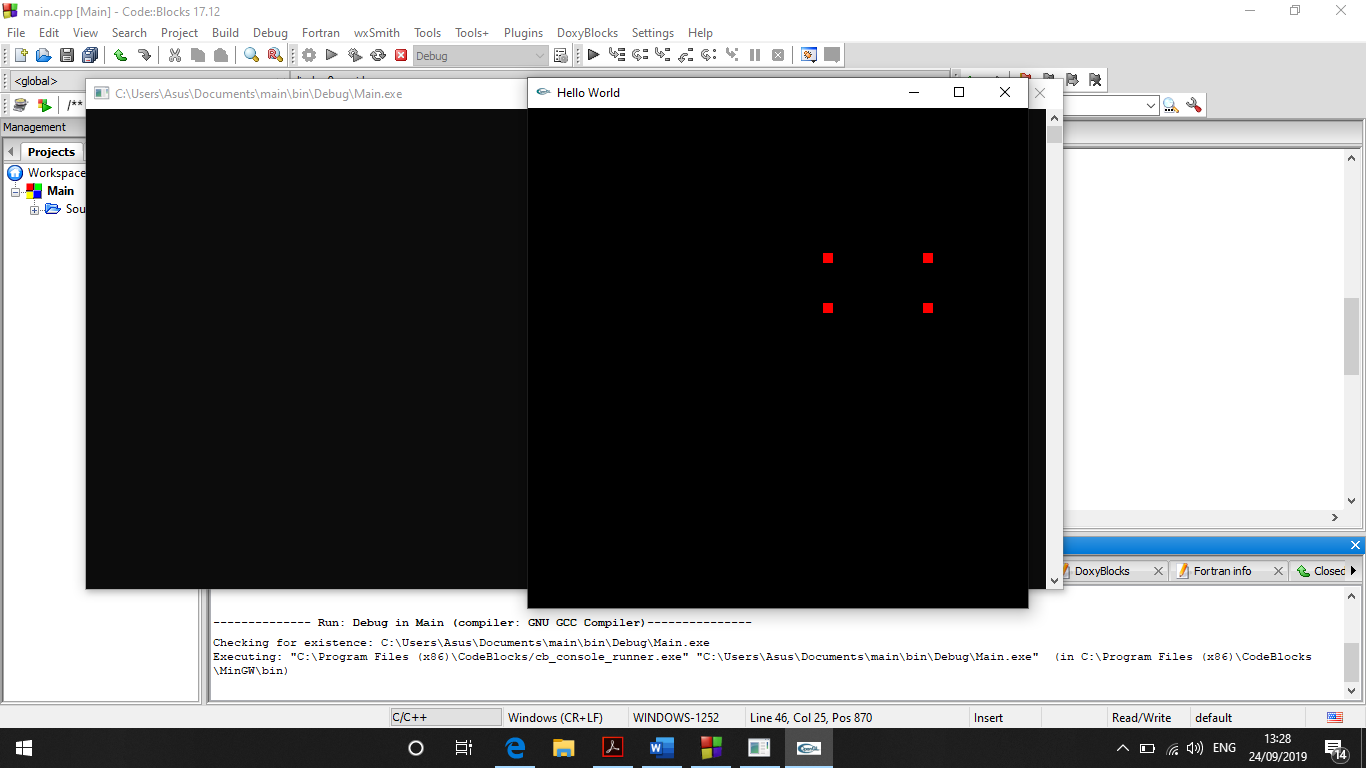
glutMainLoop();

}

Hasil

* void display(){: variable display
* glClear(GL\_COLOR\_BUFFER\_BIT);:
* glMatrixMode(GL\_MODELVIEW);:
* glPointSize(10);:
* glColor3f(1,0,0);:
* glBegin(GL\_POINTS);:
* glVertex2f(1,1);:
* glEnd();:
* gluOrtho2D(-5, 5, -5, 5);:





**Lines & Quads**

void display(){

glClear(GL\_COLOR\_BUFFER\_BIT);

glMatrixMode(GL\_MODELVIEW);

glPointSize(10);

glColor3f(1,0,0);

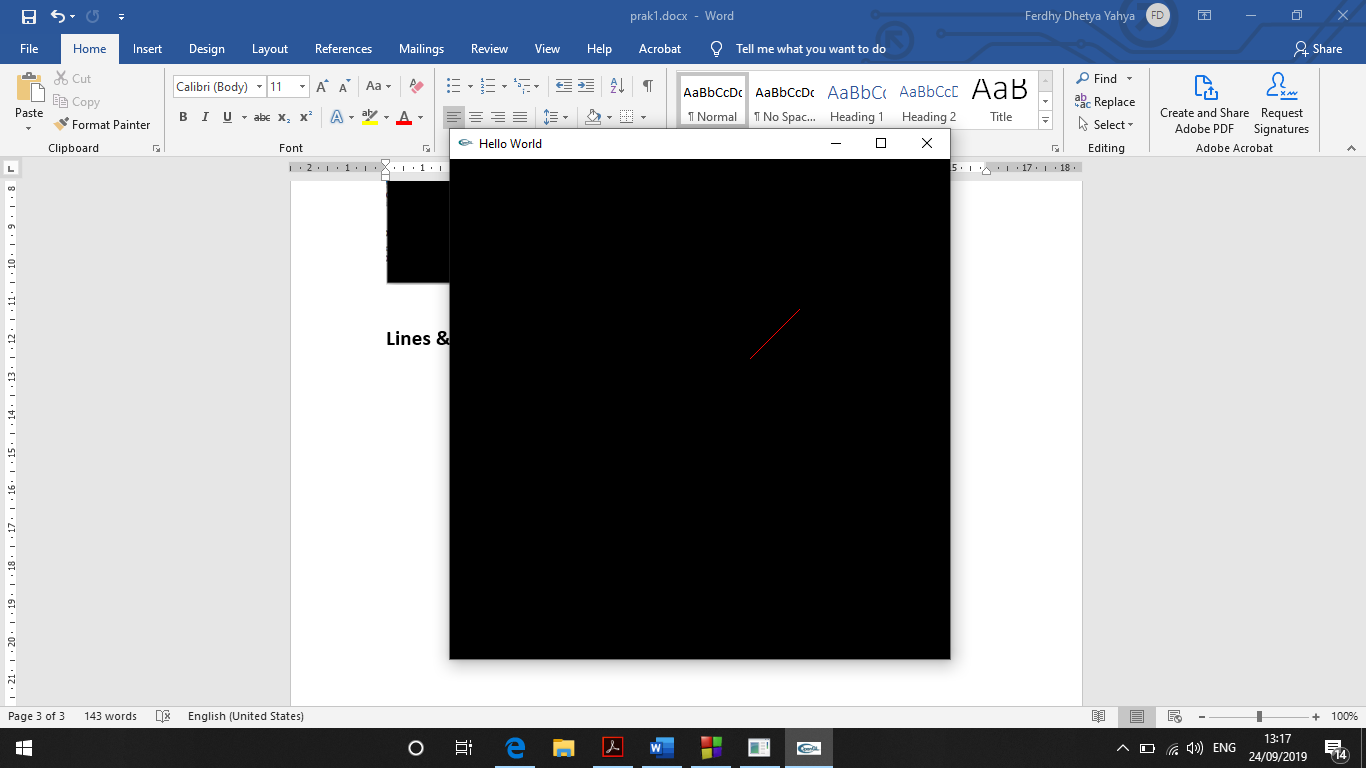
glBegin(GL\_LINES);

glVertex2f(1,1);

glVertex2f(2,2);

glEnd();

}



**Polygon**

void display(){

glClear(GL\_COLOR\_BUFFER\_BIT);

glMatrixMode(GL\_MODELVIEW);

glPointSize(10);

glColor3f(1,0,0);

glBegin(GL\_POLYGON);

glVertex2f(1,1);

glVertex2f(3,2);

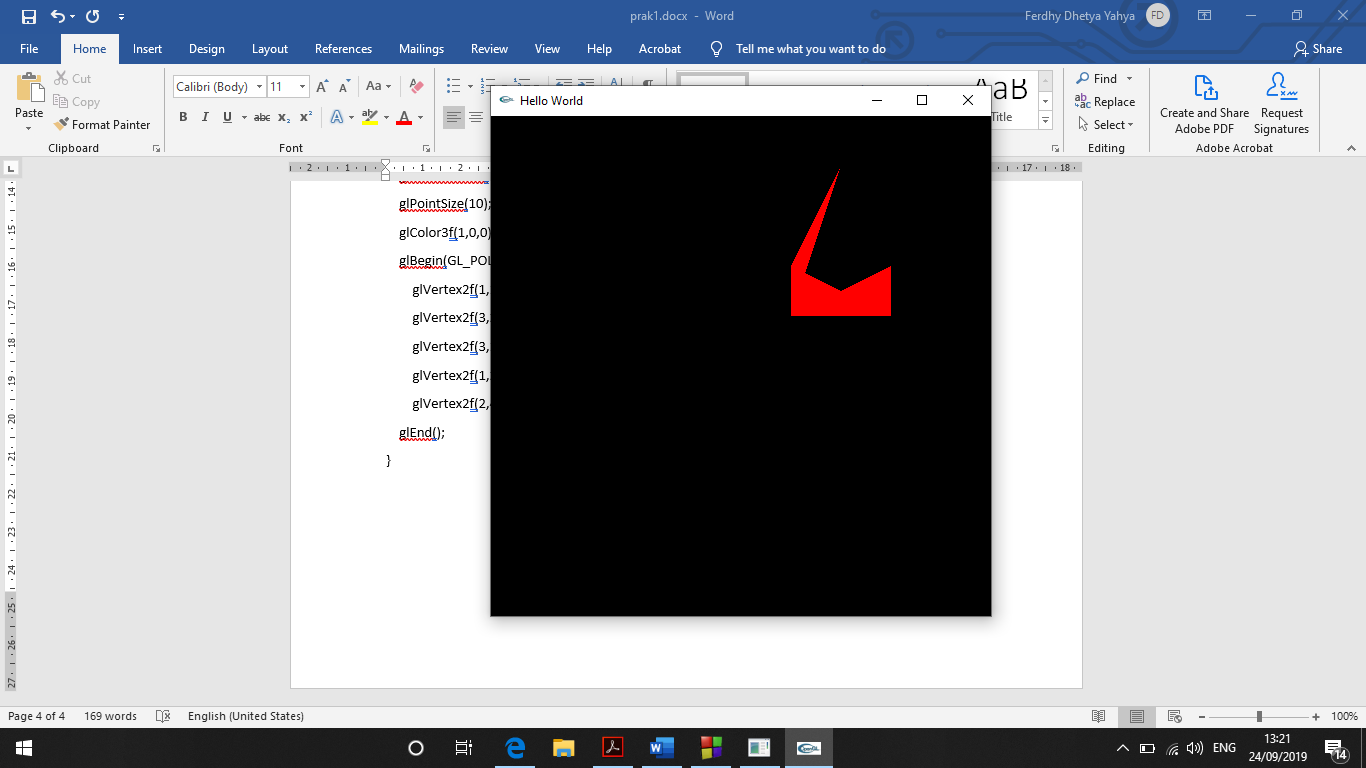
glVertex2f(3,1);

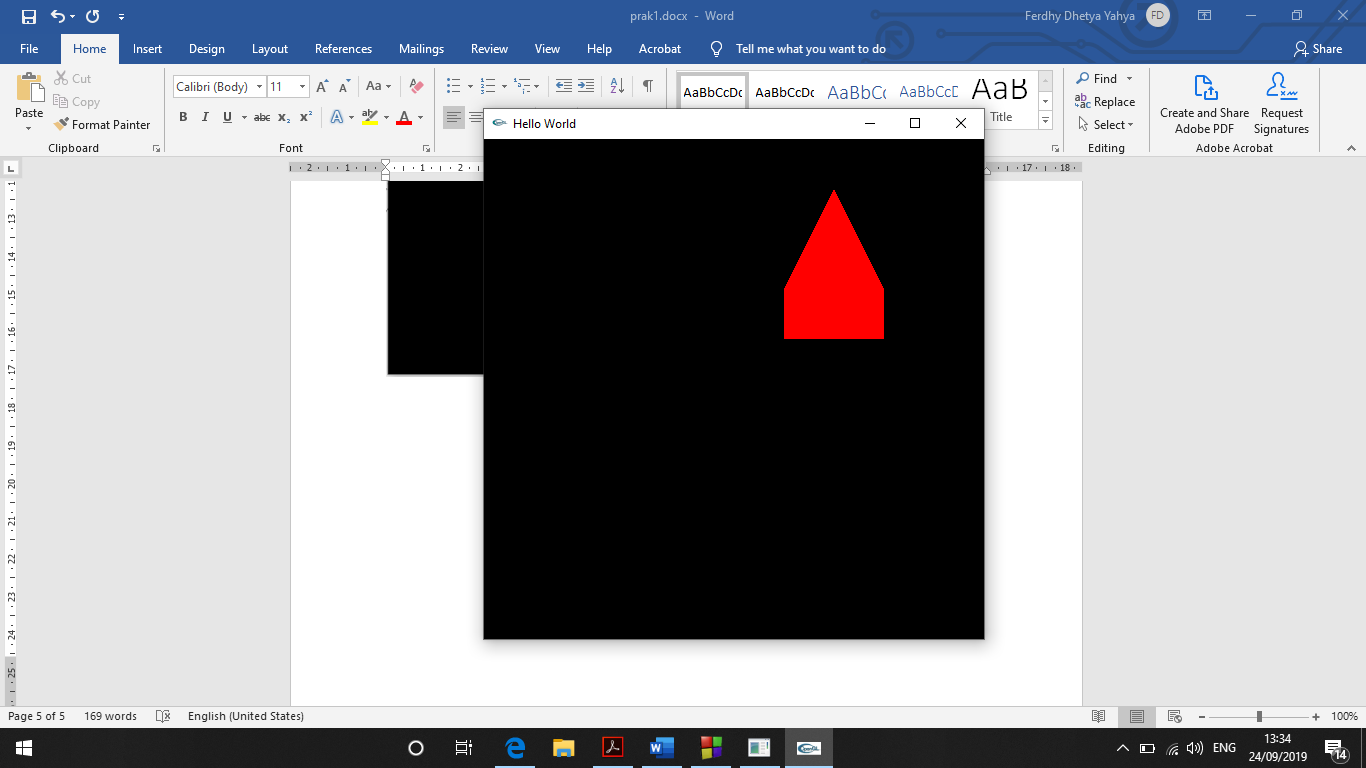
glVertex2f(1,2);

glVertex2f(2,4);

glEnd();

}





Hasil:

void display(){

glFlush();

glClear(GL\_COLOR\_BUFFER\_BIT);

glMatrixMode(GL\_MODELVIEW);

glPointSize(10);

glColor3f(1,0,0);

glBegin(GL\_POLYGON);

glVertex2f(1,1);

glVertex2f(3,1);

glVertex2f(3,2);

glVertex2f(2,4);

glVertex2f(1,2);

glEnd();