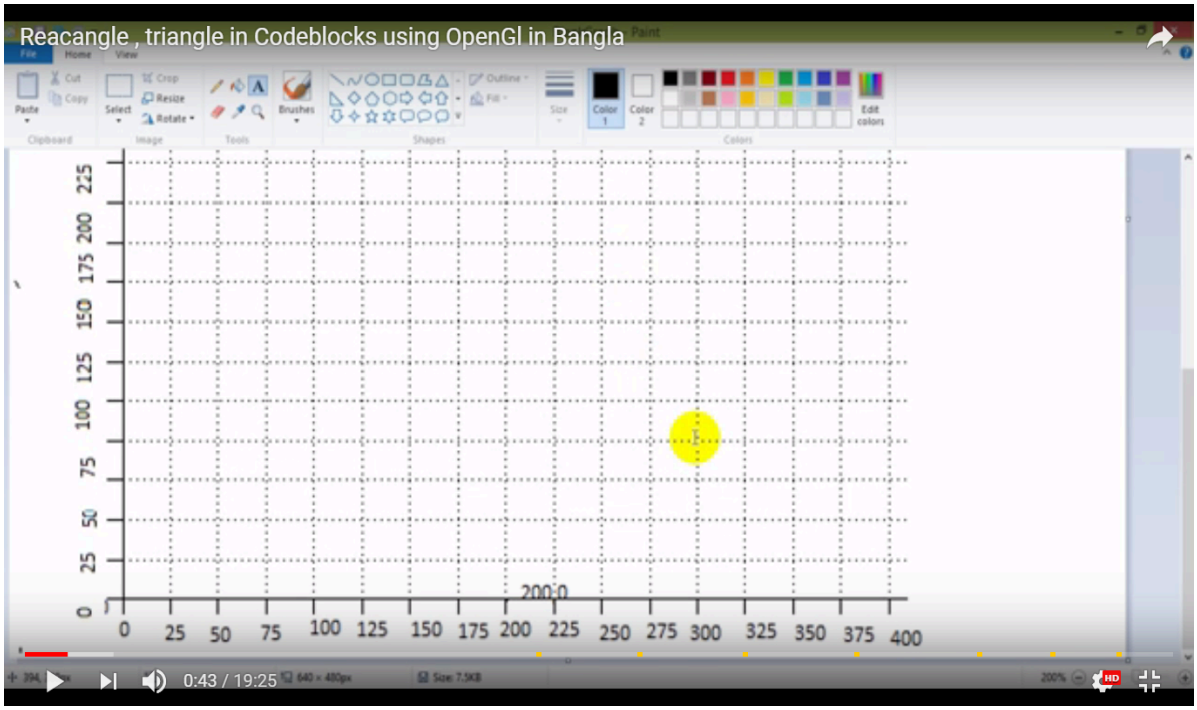
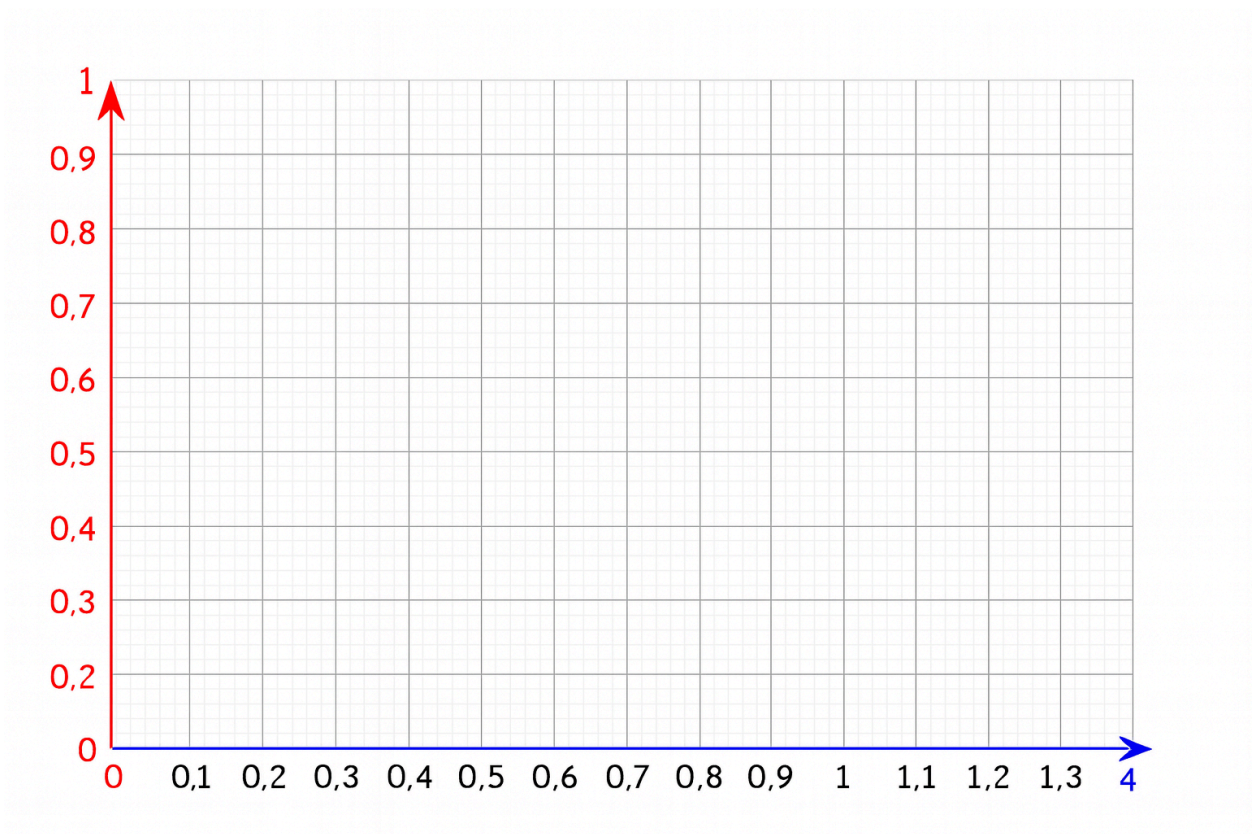
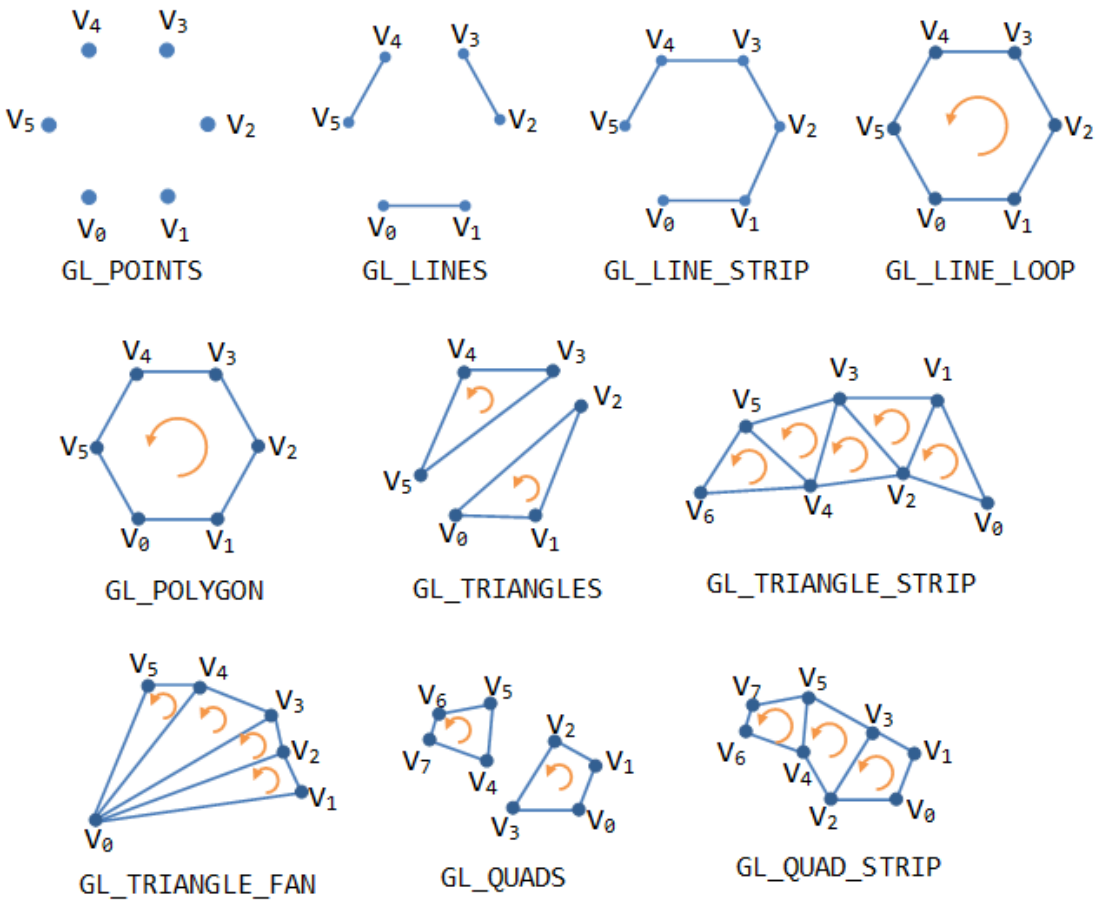

Reacangle , triangle in Codeblocks using OpenGl in Bangla
0:43 / 19:25


1
0,9
0,8
0,7
0,6
0,5
0,4
0,3
0,2
0
0
0,1
0,2
0,3
0,4
0,5
0,6
0,7
0,8
0,9
1
1,1
1,2
1,3
4



OpenGL Primitives

Some functions:

- `glutInit`: initializes GLUT, must be called before other GL/GLUT functions. It takes the same arguments as the `main()`.

```
void glutInit(int *argc, char **argv)
```

- `glutCreateWindow`: creates a window with the given title.

```
int glutCreateWindow(char *title)
```

- `glutInitWindowSize`: specifies the initial window width and height, in pixels.

```
void glutInitWindowSize(int width, int height)
```

- `glutInitWindowPosition`: positions the top-left corner of the initial window at (x, y). The coordinates (x, y), in term of pixels, is measured in window coordinates, i.e., origin (0, 0) is at the top-left corner of the screen; x-axis pointing right and y-axis pointing down.

```
void glutInitWindowPosition(int x, int y)
```

```
GL_POINTS
        GL_LINES
        GL_LINE_STRIP
        GL_LINE_LOOP
        GL_TRIANGLES
        GL_TRIANGLE_STRIP
        GL_TRIANGLE_FAN
        GL_QUADS
        GL_QUAD_STRIP
        GL_POLYGON
```

lab 1:code for Drawing line

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


void init(void)
{
        glClearColor(0.0, 0.0, 0.0, 0.0);   // Set display window colour to white

        glMatrixMode(GL_PROJECTION);                    // Set projection parameters
        gluOrtho2D(0.0, 400.0, 0.0, 400.0);
}

void drawShapes(void)
{
        glClear(GL_COLOR_BUFFER_BIT);           // Clear display window

        //Set colour to black

        glColor3f(0.0, 0.0, 0.0);
        //Adjust the point size
        glPointSize(10.0);

        // Draw a couple of points

        //Set colour to red
        glColor3f(1.0, 0.0, 0.0);
```

```
	// Draw a line
	glBegin(GL_LINES);
		glVertex2i(20, 250);
		glVertex2i(100, 80);

	glEnd();
	glFlush();		// Process all OpenGL routines
}
int main(int argc, char* argv[])
{
	glutInit(&argc, argv);                                          // Initalise GLUT
	glutInitDisplayMode(GLUT_SINGLE|GLUT_RGB); // Set display mode

	glutInitWindowPosition(100, 100);                               // Set window position
	glutInitWindowSize(350, 350);                                   // Set window size
	glutCreateWindow("An Example OpenGL Program");      // Create display window

	init();                                                         // Execute initialisation procedure
	glutDisplayFunc(drawShapes);              // Send graphics to display window
	glutMainLoop();                                 // Display everything and wait

	return 0;
}
```

Drawing triangle:

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

void init(void)
{
	glClearColor(1.0, 1.0, 1.0, 0.0);   // Set display window colour to white

	glMatrixMode(GL_PROJECTION);                    // Set projection parameters
	gluOrtho2D(0.0, 400.0, 0.0, 400.0);
}

void drawShapes(void)
{
	glClear(GL_COLOR_BUFFER_BIT);          // Clear display window

	//Set colour to black
	glColor3f(0.0, 0.0, 0.0);
	//Adjust the point size
	glPointSize(5.0);

	// Draw a couple of points
```

```
	//Set colour to blue
	glColor3f(0.0, 0.0, 3.0);

	// Draw a filled triangle
	glBegin(GL_TRIANGLES);
		glVertex2i(20, 250);
		glVertex2i(100, 380);
		glVertex2i(180, 250);
	glEnd();
glFlush();	// Process all OpenGL routines
}
```

Lab 2: home

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

void init(void)
{
	glClearColor(1.0, 1.0, 1.0, 0.0);   // Set display window colour to white

	glMatrixMode(GL_PROJECTION);			// Set projection parameters

	gluOrtho2D(0.0, 400.0, 0.0, 400.0);
}

void drawShapes(void)
{
	glClear(GL_COLOR_BUFFER_BIT);		// Clear display window

	//Set colour to black
	glColor3f(0.0, 0.0, 0.0);
	//Adjust the point size
	glPointSize(5.0);

	// Draw a couple of points

	//Set colour to green
	glColor3f(0.0, 0.0, 3.0);

	// Draw a filled triangle
	glBegin(GL_TRIANGLES);
		glVertex2i(20, 250);
		glVertex2i(100, 380);
		glVertex2i(180, 250);
```

```
glEnd();

//Set colour to red
glColor3f(0.0, 5.0, 0.0);

// Draw a filled quadrilateral
glBegin(GL_QUADS);
        glVertex2i(200, 250);
        glVertex2i(200, 380);
        glVertex2i(380, 380);
        glVertex2i(380, 250);
glEnd();

/*//Set colour to blue
glColor3f(0.0, 0.0, 1.0);

// Draw a filled octagon
glBegin(GL_POLYGON);
        glVertex2i(90, 30);
        glVertex2i(30, 90);
        glVertex2i(30, 174);
        glVertex2i(90, 234);
        glVertex2i(174, 234);
        glVertex2i(234, 174);
        glVertex2i(234, 90);
        glVertex2i(174, 30);
glEnd();

//Set colour to black
glColor3f(0.0, 1.0, 1.0);

// Draw an outlined triangle
glBegin(GL_LINES);
        glVertex2i(100, 200);
        glVertex2i(100, 50);
        glVertex2i(100, 50);
        glVertex2i(300, 50);
        glVertex2i(300, 50);
        glVertex2i(300, 200);
        glVertex2i(300, 200);
        glVertex2i(100, 200);



glEnd();
*/

glFlush();          // Process all OpenGL routines
```

```
}

int main(int argc, char* argv[])
{
	glutInit(&argc, argv);                                          // Initalise GLUT
	glutInitDisplayMode(GLUT_SINGLE|GLUT_RGB); // Set display mode

	glutInitWindowPosition(50, 100);                                // Set window position
	glutInitWindowSize(400, 300);                                   // Set window size
	glutCreateWindow("An Example OpenGL Program");      // Create display window

	init();                                                         // Execute initialisation procedure
	glutDisplayFunc(drawShapes);              // Send graphics to display window
	glutMainLoop();                                         // Display everything and wait

	return 0;
}
```