Kingdom of Saudi Arabia
المملكة العربية السعودية
Ministry of Education
وزارة التعليم
Umm AlQura University
جامعة أم القرى
Adham University College
الكلية الجامعية بأضم
Computer Science Department
قسم الحاسب الآلي

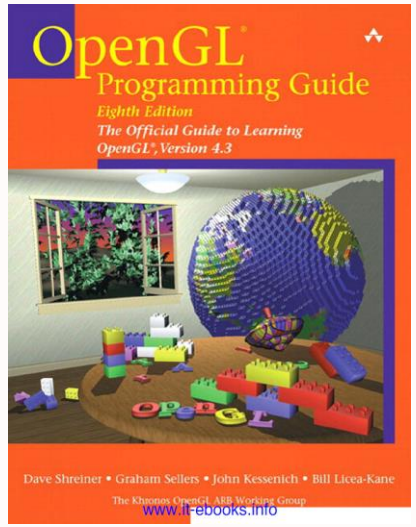
CS Department

Computer Graphics Course, 3-6803430

T.Mariah Khayat

References

- *Lab Lectures, Computer Graphics, Taif University, Faculty Of Computers And Information Technology, TA. Maha Thafar &TA. Haifa Alshehri, TA.Sohair Soliman & L.Shakila Bano.*
- *OpenGL Programming Guide: The Official Guide to Learning OpenGL, Versions 4.3, 8th edition, Dave Shreiner, Graham Sellers, John Kessenich, Bill Licea-Kane & The Khronos OpenGL ARB Working Group, Addison-Wesley.*



Lecture Three

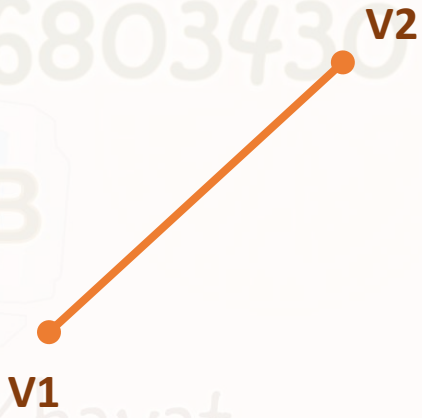
Drawing Lines using OpenGL

content

1. Line
2. Draw a Line using OpenGL
3. Stipple Line
4. Draw a stipple line using OpenGL
5. Draw a strip line using OpenGL

Lines

- **A Line:** is a straight line in space.
- It is defined by its two end-points.
- Each end-point is called a ***vertex*.**
- Each vertex has its own 3-D/2-D position defined by its 3-D/2-D coordinates.
- We draw lines by sampling at intervals of one pixel and drawing the closest pixels.



Drawing a Line Using OpenGL

```
#include<windows.h>
#include<GL/glut.h>
void DrawLine(){
 glClear(GL_COLOR_BUFFER_BIT);
 glLineWidth(8);
 glBegin(GL_LINES);
 glVertex2d(400, 700);
 glVertex2d(100, 50);
 glEnd();
 glFlush();
}
void Initial(){
  glClearColor(0.0, 1.0, 1.0, 0.0);
  glColor3f(0,1,0);
  glMatrixMode(GL_PROJECTION);
  glLoadIdentity();
  gluOrtho2D(0, 600, 0, 800);
}
int main(){
glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB);
glutInitWindowSize(400, 400);
glutInitWindowPosition(0,0);
glutCreateWindow("Draw a Line");
Initial();
glutDisplayFunc(DrawLine);
glutMainLoop();
return 0;
}
```

Drawing a Line Using OpenGL

Draw a Line.cpp [Draw a Line] - Code::Blocks 17.12

File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings Help

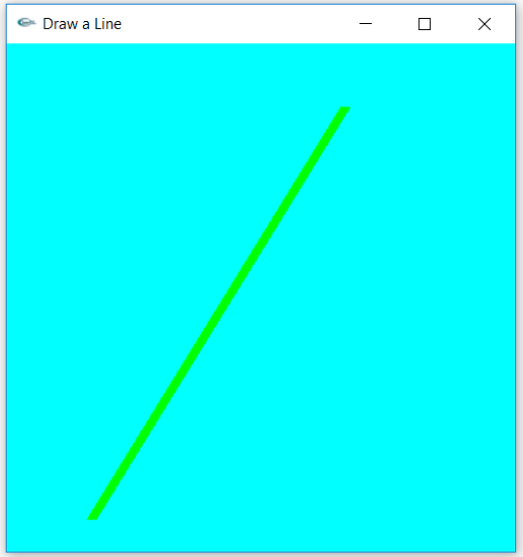
Management

Projects Symbols Files

Workspace
Draw a Line
Sources
Draw a Line.cpp
main.c

Draw a Line.cpp

```
#include<windows.h>
#include<GL/glut.h>
void DrawLine(){
 glClear(GL_COLOR_BUFFER_BIT);
 glLineWidth(8);
 glBegin(GL_LINES);
 glVertex2d(400, 700);
 glVertex2d(100, 50);
 glEnd();
 glFlush();
}
void Initial(){
  glClearColor(0.0, 1.0, 1.0, 0.0);
  glColor3f(0,1,0);
  glMatrixMode(GL_PROJECTION);
  glLoadIdentity();
  gluOrtho2D(0, 600, 0, 800);
}
int main(){
glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB);
glutInitWindowSize(400, 400);
glutInitWindowPosition(0,0);
glutCreateWindow("Draw a Line");
Initial();
glutDisplayFunc(DrawLine);
glutMainLoop();
return 0;
}
```



Logs & others

Code::Blocks | Search results | Cccc | Build log | Build messages | CppCheck/Vera++ | CppCheck/Vera++ messages | Cscope | Debugger | DoxyBlocks | Fortran info

```
\main.c" -o obj\Debug\main.o
mingw32-g++.exe  -o "bin\Debug\Draw a Line.exe" "obj\Debug\Draw a Line.o" obj\Debug\main.o   -lopengl32 -lglu32 -lgdi32 "C:\Program Files (x86)\CodeBlocks\MinGW\lib\glut32.lib"
Output file is bin\Debug\Draw a Line.exe with size 48.22 KB
Process terminated with status 0 (0 minute(s), 0 second(s))
0 error(s), 0 warning(s) (0 minute(s), 0 second(s))
```

C/C++ | Windows (CR+LF) | default | Line 6, Col 16, Pos 125 | Insert | Read/Write | default

Stipple Line

- To determine the type /stipple of line use 3 functions:

```
1.glEnable(GL_LINE_STIPPLE);
2.glLineStipple(GLint factor , GLushort pattern);
3.glDisable(GL_LINE_STIPPLE);
```

- **First:** `glEnable(GL_LINE_STIPPLE)` Function:
  - To make active line stipple.
- **Second:** `glLineStipple` Function:
  - **Example:** `glLineStipple(2, 0x3f07);`
    - Parameter 1: **Factor:**

*Specifies a multiplier for each bit in the line stipple pattern. In above example factor =2 means: each bit in the pattern is used 2 times before the next bit in the pattern is used.
*Factor is clamped to the range [1, 256] and defaults to 1.

Stipple Line

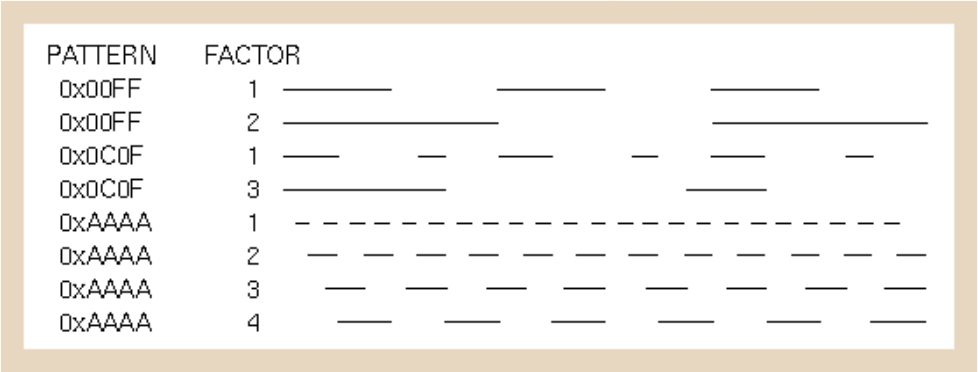
- **Second: `glLineStipple` Function:**
  - Parameter 2: **Pattern:**

*Specifies in hexadecimal transform to binary number (16-bit integer) whose bit pattern determines which fragments of a line will be drawn

*0 is OFF , 1 is ON

*Bit zero is used first; the default pattern is all 1s.

*pattern **= `0x3f07 --------> 0011111100000111`**



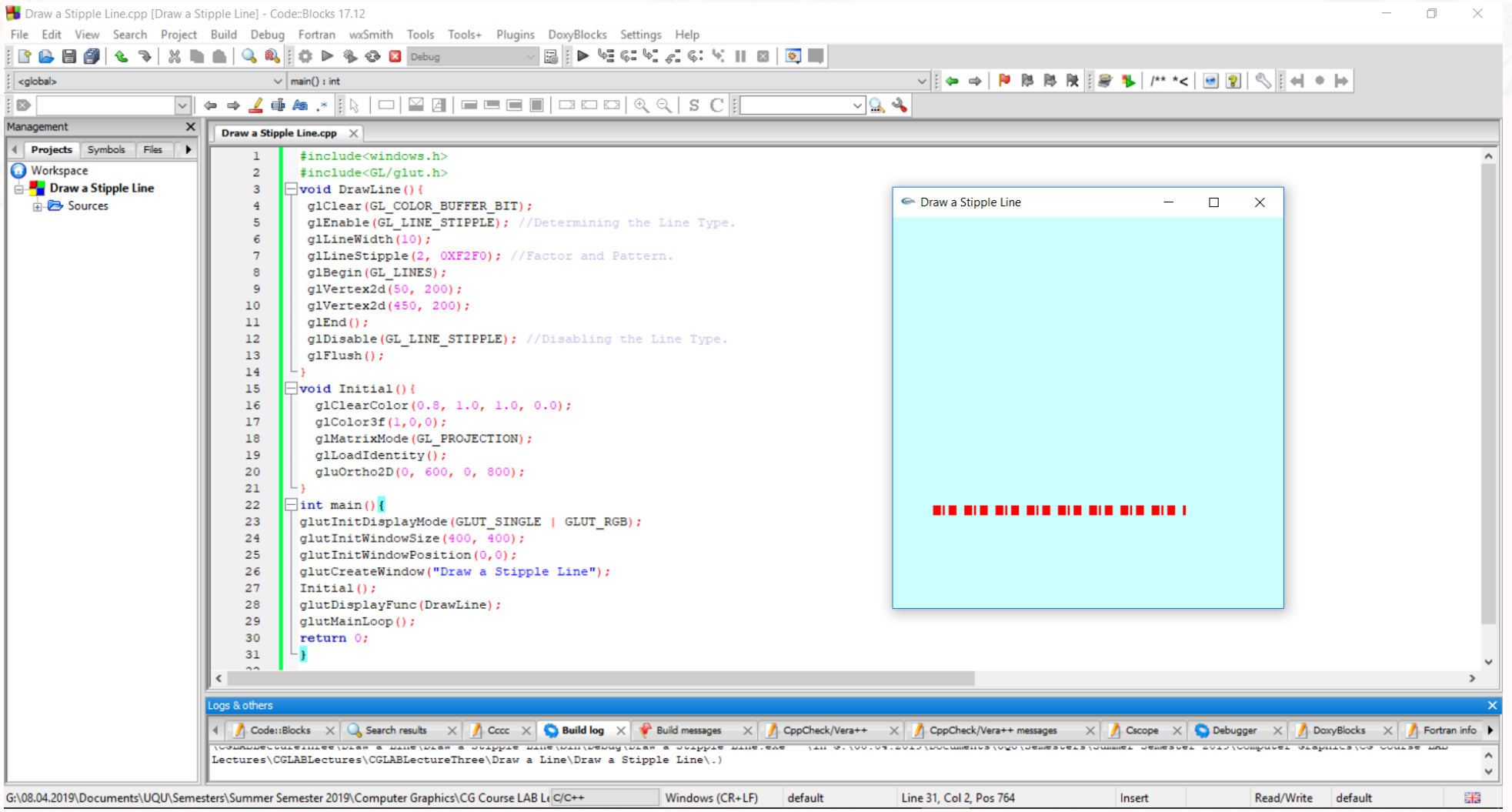
Stipple Line

- **Third : `glDisable(GL_LINE_STIPPLE)`** Function:
  - To stop active (close) line stipple.

Drawing Stipple Line Using OpenGL

```
#include<windows.h>
#include<GL/glut.h>
void DrawLine(){
 glClear(GL_COLOR_BUFFER_BIT);
 glEnable(GL_LINE_STIPPLE); //Determining the Line Type.
 glLineWidth(10);
 glLineStipple(2, 0XF2F0); //Factor and Pattern.
 glBegin(GL_LINES);
 glVertex2d(50, 200);
 glVertex2d(450, 200);
 glEnd();
 glDisable(GL_LINE_STIPPLE); //Disabling the Line Type.
 glFlush();
}
void Initial(){
  glClearColor(0.8, 1.0, 1.0, 0.0);
  glColor3f(1,0,0);
  glMatrixMode(GL_PROJECTION);
  glLoadIdentity();
  gluOrtho2D(0, 600, 0, 800);
}
int main(){
glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB);
glutInitWindowSize(400, 400);
glutInitWindowPosition(0,0);
glutCreateWindow("Draw a Stipple Line");
Initial();
glutDisplayFunc(DrawLine);
glutMainLoop();
return 0;
}
```

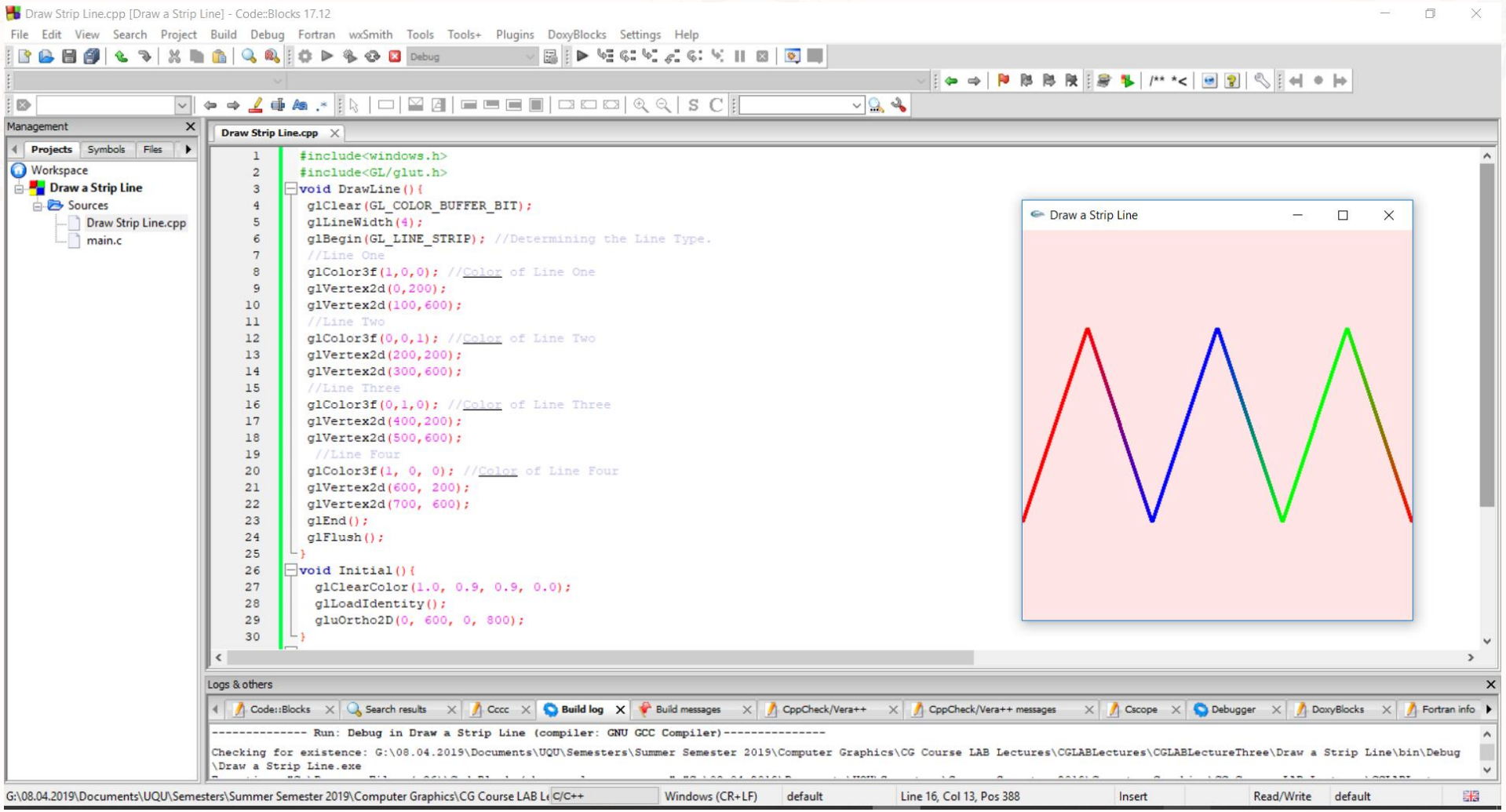
Drawing Stipple Line Using OpenGL



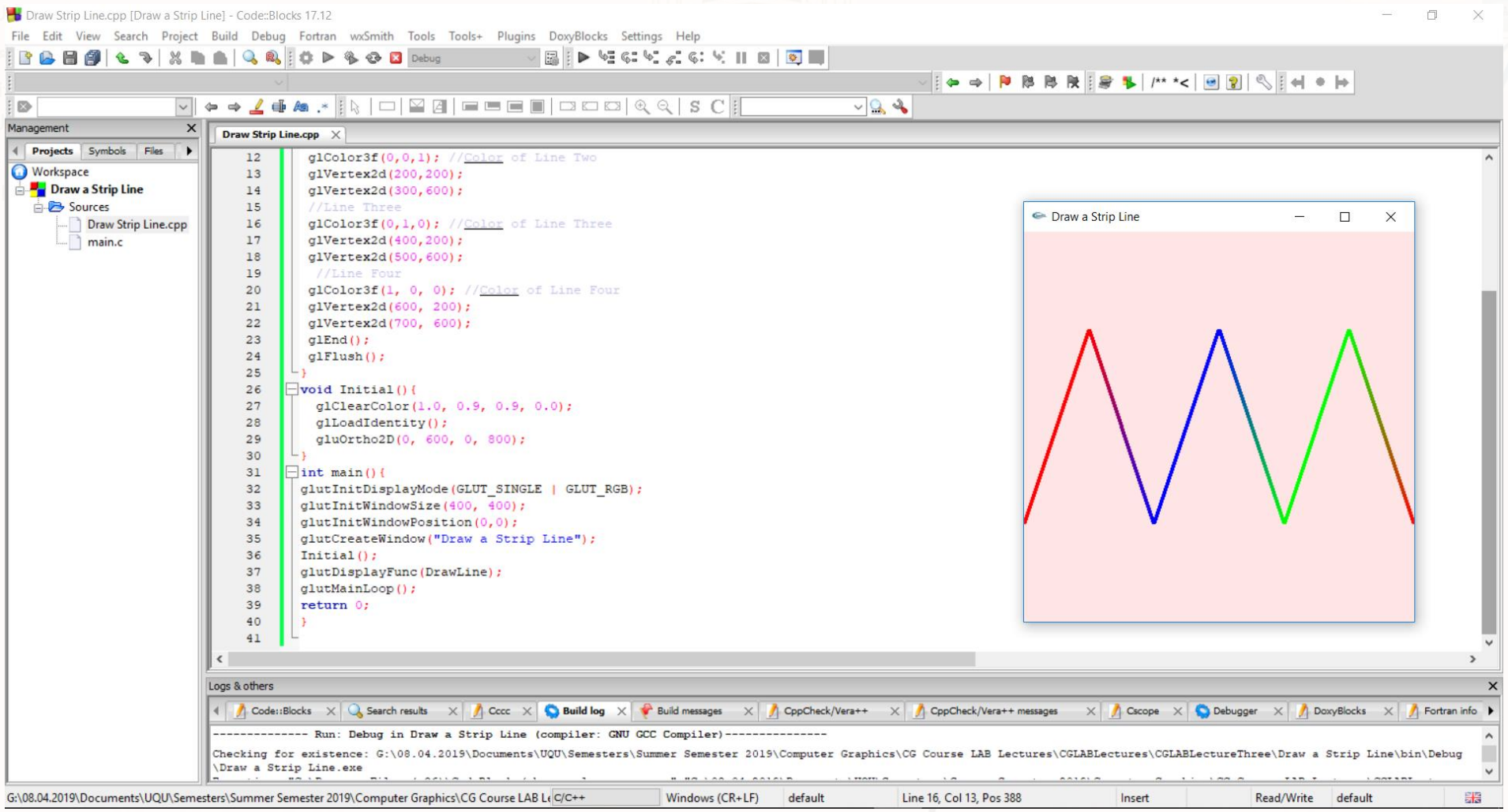
Drawing Strip Line Using OpenGL

```
#include<windows.h>
#include<GL/glut.h>
void DrawLine(){
 glClear(GL_COLOR_BUFFER_BIT);
 glLineWidth(4);
 glBegin(GL_LINE_STRIP); //Determining the Line Type.
 //Line One
 glColor3f(1,0,0); //Color of Line One
 glVertex2d(0,200);
 glVertex2d(100,600);
 //Line Two
 glColor3f(0,0,1); //Color of Line Two
 glVertex2d(200,200);
 glVertex2d(300,600);
 //Line Three
 glColor3f(0,1,0); //Color of Line Three
 glVertex2d(400,200);
 glVertex2d(500,600);
  //Line Four
 glColor3f(1, 0, 0); //Color of Line Four
 glVertex2d(600, 200);
 glVertex2d(700, 600);
 glEnd();
 glFlush();
}
void Initial(){
  glClearColor(1.0, 0.9, 0.9, 0.0);
  glLoadIdentity();
  gluOrtho2D(0, 600, 0, 800);
}
int main(){
glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB);
glutInitWindowSize(400, 400);
glutInitWindowPosition(0,0);
glutCreateWindow("Draw a Strip Line");
Initial();
glutDisplayFunc(DrawLine);
glutMainLoop();
return 0;
}
```

Drawing Strip Line Using OpenGL



Drawing Strip Line Using OpenGL



Exercise

- **Draw a Stipple Line Based on the following specifications:**
  - **Line Width = 5.**
  - **Factor = 1, Pattern = 0XAAAA.**
  - **Vertex 1 = (200, 400) and Vertex 2 = (500, 400).**
  - **Line Color = White.**
  - **Background Color = Red.**
  - **Window Title Bar = “My White Stipple Line”.**

وصلى الله وبارك على نبينا محمد

The End Summary of Lecture Three

T.Mariah Khayat
الأستاذة/ مارية خياط
Adham University College
الكلية الجامعية بأضم

mskhayat@uqu.edu.sa