

# Lab 5: District CG

You are given a plot of land in district CG and given the freedom to build whatever you want on it, be it a house, a farm, a school, a church, a temple, a stadium, etc.

However, due to the shortage of land and resources, you are only given one square plot, and your construction materials limited to a set of 40 items. To start building on your plot of land, please download lab5.zip from the IVLE workbin.

## Building Instructions

Once you run `lab5.exe`, you will see all the 81 plots of land. However, you can only build on one plot. Trespassers will be prosecuted! To begin, the dimensions of your plot of land is  $8 \times 8 \times 12$  (length x breadth x height). This volume is drawn as a wireframe cube in the program. Therefore, please construct your building in `lab5.cpp` such that its coordinates lies within

- ☐  $-4 < x < 4$
- ☐  $0 < y < 12$
- ☐  $-4 < z < 4$

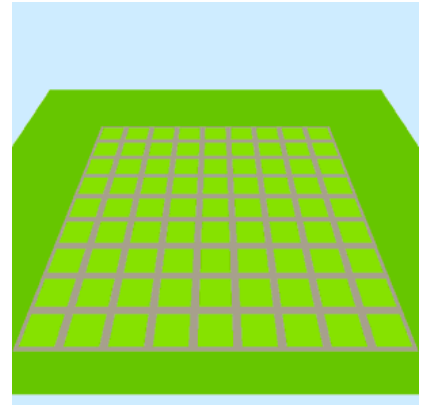
Upon opening `la5.cpp`, you will see a function `drawMyHouse()` that draws the bounding wireframe cuboid. Please rename the function `drawMyHouse()` to a function name with the following format `{your matric number}House()` eg `U012345House()`. Note that it's a capital U, and you don't include the last alphabet in your matriculation number. Similarly, update the function name `drawMyHouse()` in `drawHouses.h` with your new function name. Once you are done, you can start adding your code into the function to draw your house. If you need to create more functions other than `{matric number}House()`, please name your function `{matric number}{function name}()`.

To preview your building on the plot, go to `main.cpp` and change `drawMyHouse()` in `display()` to your function name. You may wish to observe your building up close by right-clicking on your plot of land. Dragging the left mouse button allows you to rotate the view. Return to the full view of District CG using the middle click.

## Construction Materials

Due to limited resources, District CG only has access to 40 types of construction materials, whose samples are given in the appendix. These materials are pre-loaded into program in the `init()` function. You may use whichever material you wish to texture your building.

```
texSet[0] = loadMyTextures("brick1.bmp");
texSet[1] = loadMyTextures("brick2.bmp");
texSet[2] = loadMyTextures("brick3.bmp");
texSet[3] = loadMyTextures("brick4.bmp");
texSet[4] = loadMyTextures("door1.bmp");
texSet[5] = loadMyTextures("door2.bmp");
...
```



To use them, make your function above take in the array of texture eg `void drawMyHouse (GLuint texSet[])`. Next, when calling your function in `display()`, pass the texture set into your function just like `drawMyHouse (texSet)`. To use a certain texture on a polygon, you first need to enable 2D texture. Next bind the desired texture (eg texture 0 in the following example).

```
glEnable(GL_TEXTURE_2D);  
glBindTexture( GL_TEXTURE_2D, texSet[0] );
```

Next, define the texture coordinates for each vertex.

```
glBegin(GL_QUADS);  
glTexCoord2d(0.0,0.0);glVertex3f(-3, 0, -1.5);  
glTexCoord2d(1.0,0.0);glVertex3f(3, 0, -1.5);  
glTexCoord2d(1.0,1.0);glVertex3f(3, 3, -1.5);  
glTexCoord2d(0.0,1.0);glVertex3f(-3, 3, -1.5);  
glEnd();
```

## Frame rate requirement

We are ultimately going to combine all your plots into one huge plot for the course. Therefore, please ensure that you do not have complex geometry and texturing that requires too much time to render. As a guideline, your drawing should be able to maintain a frame rate of at least 100 frames per second. A frame rate counter has been provided for you, which prints the frame rate on the console window.


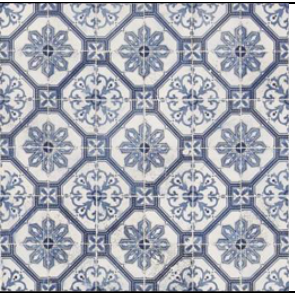



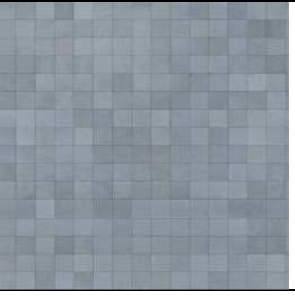
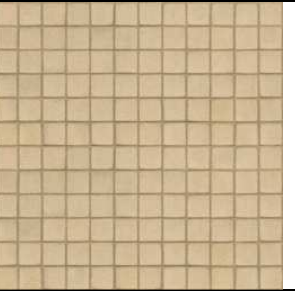

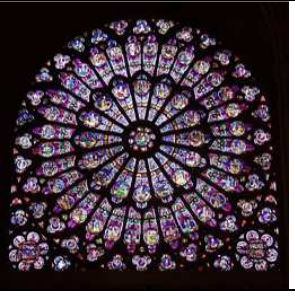

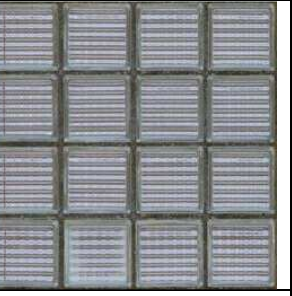
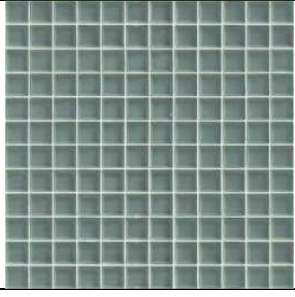







## Submission Instructions

For submission, only submit your .cpp file and **rename** it to “*{your matric number}.cpp*”. One final note for this assignment, everything is graded in debug mode (meaning, you don’t have to care about the release mode).

## Appendix

			
[texID=0]Brick1	[texID=1]Brick2	[texID=2]Brick3	[texID=3]Brick4
			
[texID=4]Door1	[texID=5]Door2	[texID=6]Flower1	[texID=7]Flower2
			
[texID=8]Flower3	[texID=9]Grass1	[texID=10]Grass2	[texID=11]Leaves1
			
[texID=12]Leaves2	[texID=13]Roof1	[texID=14]Roof2	[texID=15]Roof3
			
[texID=16]Roof4	[texID=17]Roof5	[texID=18]Roof6	[texID=19]Stone1



			
[texID=20]Stone2	[texID=21]Tile1	[texID=22]Tile2	[texID=23]Tile3
			
[texID=24]Tile4	[texID=25]Tile5	[texID=26]Tile6	[texID=27]Window1
			
[texID=28]Window2	[texID=29]Window3	[texID=30]Window4	[texID=31]Window5
			
[texID=32]Window6	[texID=33]Wood1	[texID=34]Wood2	[texID=35]Wood3
			
[texID=36]Wood4	[texID=37]Wood5	[texID=38]Wood6	[texID=39]Wood7