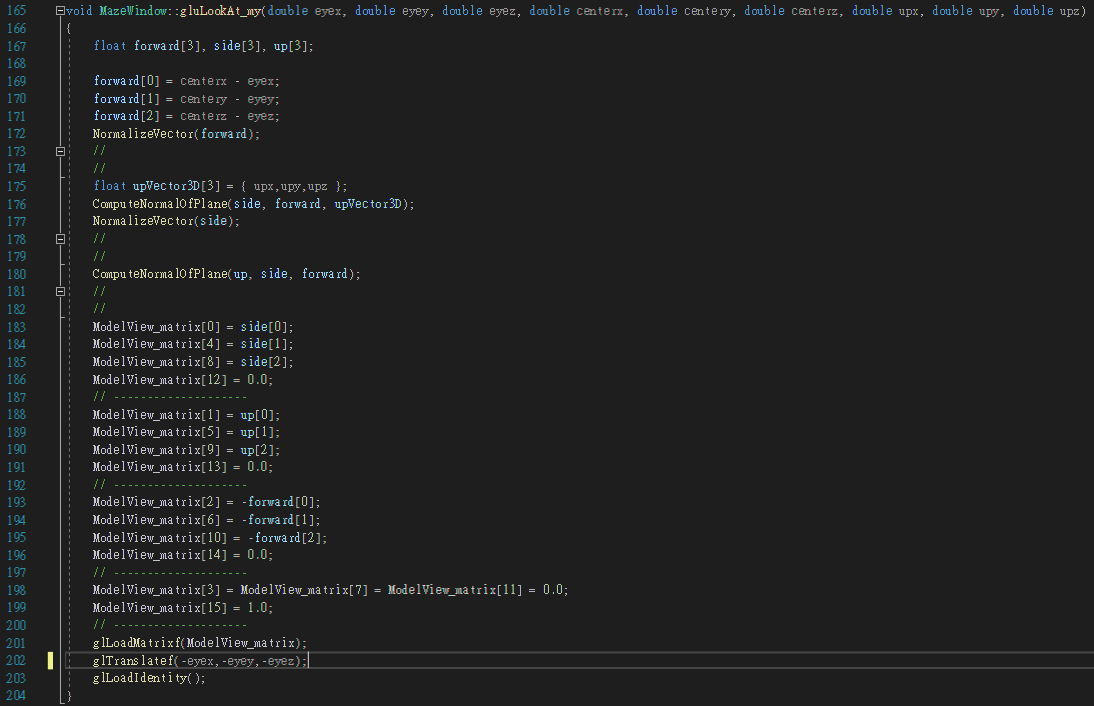
整體來說是將Vertex先乘上ModelView matrix(gluLookAt)，再乘projection matrix(gluPerspective)到齊次座標後，再clip他們，只畫出有在視野內的牆，而後再將vertex3f捨棄，改以vertex2f交給openGL繪出。

ModelView\_matrix:

生成位置:MazeWindow的gluLookAt\_my。

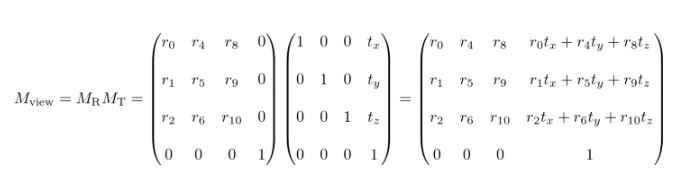
而這個Viwe\_Matrix在MazeWindow及Maze中都會用到，故將其設為MazeWindow的靜態成員。



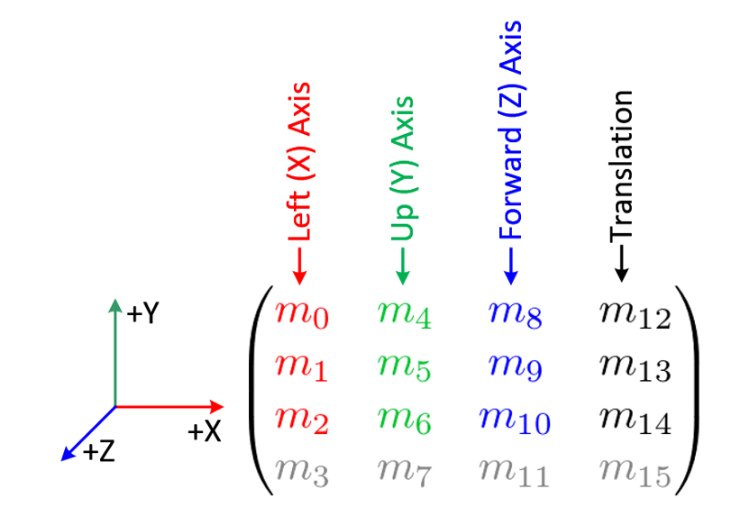
傳入參數: (eye , center , up)



而ModelView\_Matrix = Rotate \* Translate



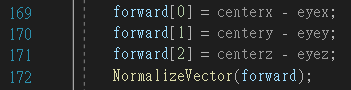
而根據定義可以將ModelView\_Matrix理解為



而其中Translate是將所有座標上的點跟著相機一起加上向量(-eyex,-eyey,-eyez)平移至(0,0,0)，所以是 (-eyex,-eyey,-eyez)。

接下來三個，分別為Left、Up、Forward，對應到我的函式為Side、Up、Forward。

Forward向量為相機位置(eye)射向相機看著的點(center)，所以為



\*其中NormalizeVector 

\*而目前的Forward為z軸。

定接下來求Left及Up(希望指向天空)，可以利用up of world space，也就是傳進來函式的參數(up,,upy,upz)，通常為world space的Y軸(0,1,0)。

world space的Y軸: 

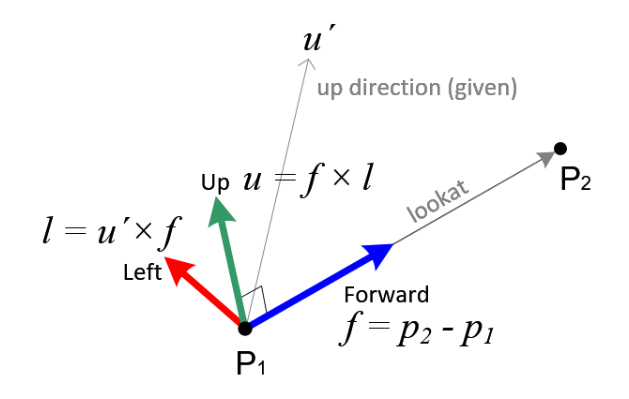
可以使Y軸與Forward外積得到Left(-x)。



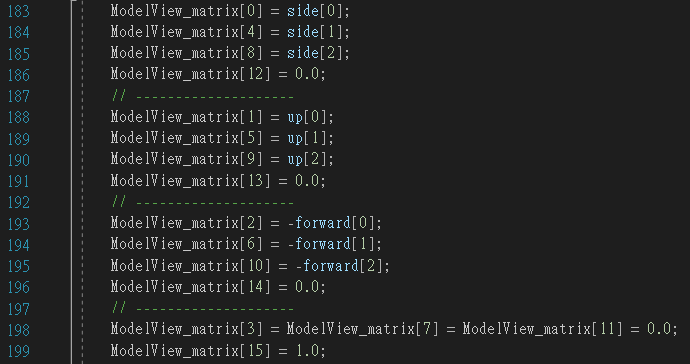
\*其中ComputeNormalOfPlane (向量C ,向量A,向量B): C = A cross B;

再以Forward(z)及Left(-x)外積後即得到camara Space的Y軸，也就是Up。



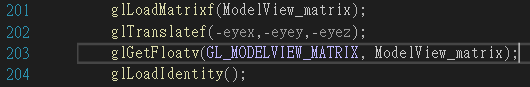


形成Rotate:



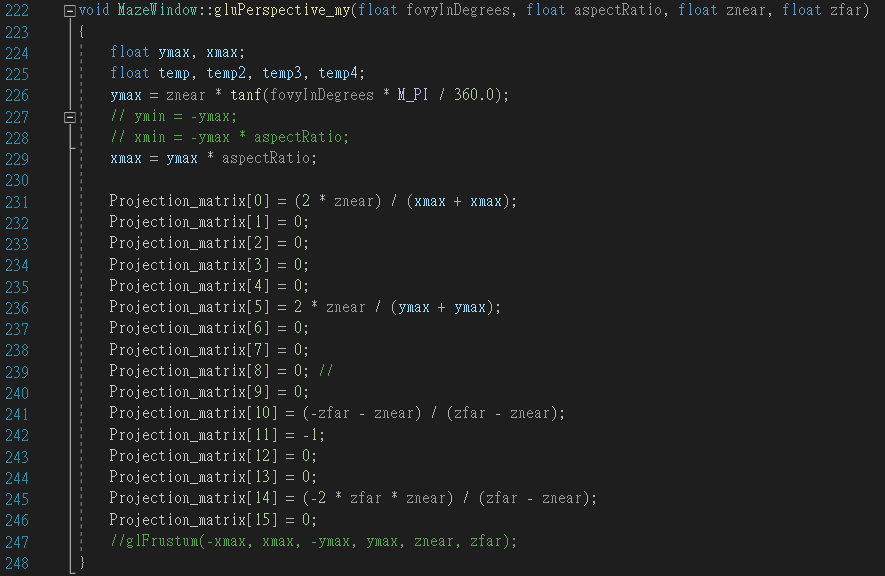
\*其中forward要乘負號原因為指向為正Z軸，而相機眼前的朝向為-Z軸。

再將此Rotate乘上Translate完成ModelView\_Matrix :

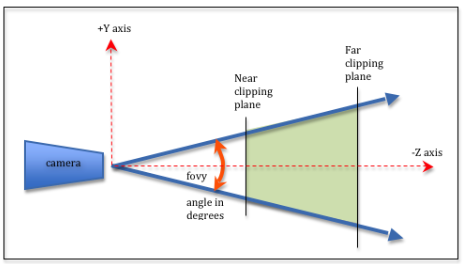


Projection\_Matrix:

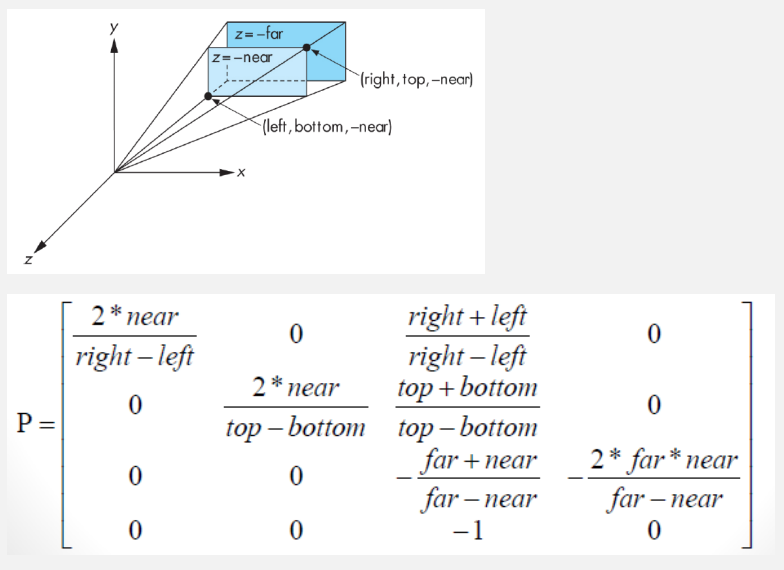
生成位置:MazeWindow的gluPerspective\_my。



參數示意:



基本上是照著這個填的:



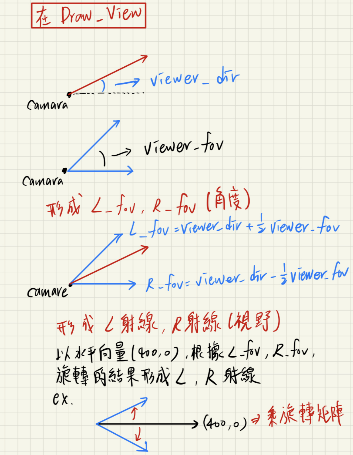
而其中right = xmax , Left = -xmax(視野左右對稱)

Top = ymax, bottom = -ymax。

Clip :

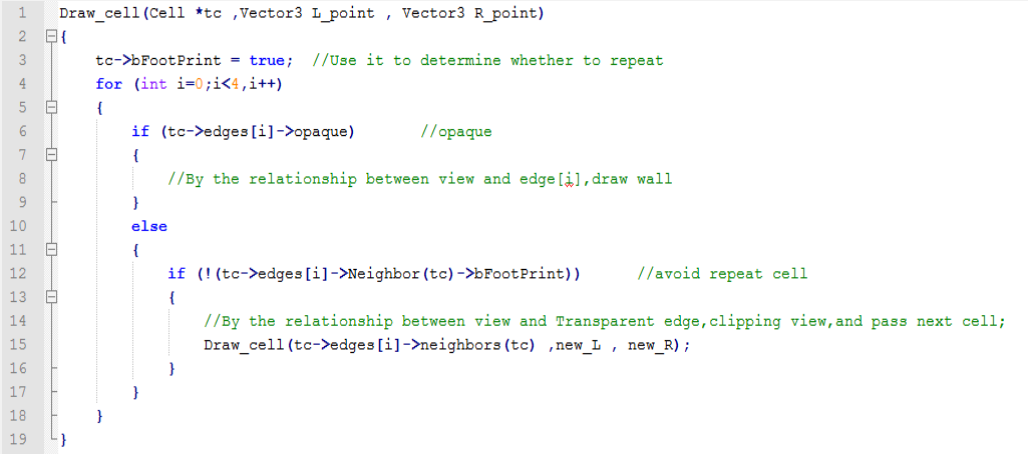
在Draw\_View:

生成初始左視野射線以及右視野射線



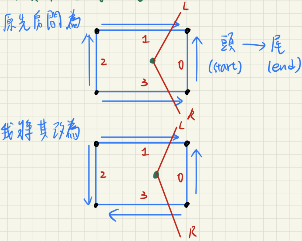
而後進入Draw\_Cell

在Draw\_Cell:

以此種遞回方式實作

在進入for(四道牆)的迴圈後、判斷透不透明前會 :

1. 交換特定牆的頭(start)跟尾(end)

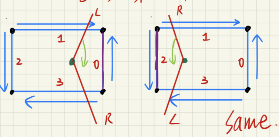


對牆0、2來說，若R與其相交，則都是將其牆頭更新為交點。

若L與其相交，則都是將其牆尾更新為交點。

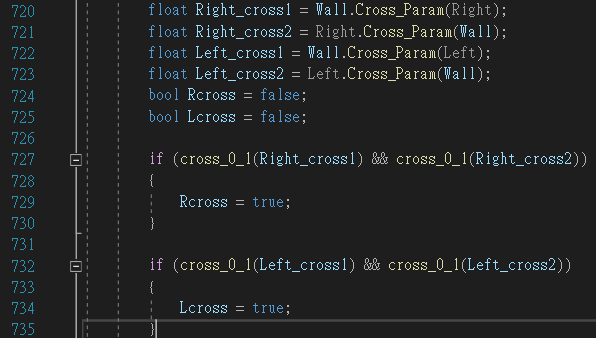
對牆1、3來說，若R與其相交，則都是將其牆尾更新為交點。

若L與其相交，則都是將其牆頭更新為交點。



上圖為兩間相鄰房間，紫色牆為同一面透明牆，可見面朝右、朝左去切這道透明牆，效果都是合理的。

1. 判斷與這道牆與R、L射線分別有沒有交。



可用於後續判斷此牆有沒有在視野內。

\*判斷牆有無在視野內:

1.與射線有交 :Rcross或 Lcross為true。(相交型)

2.若Rcross 及 Lcross 皆 false，則看此牆是否為完整在視野內。

也就是牆頭牆尾皆在右射線的左邊and皆在左射線的右邊。

(全在視野內型)

\*判斷點與線的關係:



在射線右邊:回傳1

在射線上:回傳0

在射線左邊:回傳-1

而後判斷透or不透明:

若牆為opaque:

初始設定draw\_edge\_start = 牆的start

Draw\_edge\_end = 牆的 end

Insight = false

If(Rcross or Lcross) 相交型

Insight = true

If(Rcross)

根據牆號去將交點打入draw\_edge\_start或是end

If(Lcross)

根據牆號去將交點打入draw\_edge\_start或是end

Else 可能全在視野內型

判斷有沒有全在視野內

有->不用更新draw\_edge\_start及end, Insight = true

沒有->無事發生

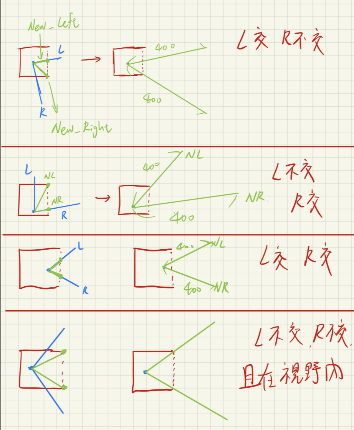
If(Insight)

畫draw\_edge

若牆為不透明:

初始設定new\_Right = Right

New\_Left = Left



若此隱形牆在視野內，則會遞迴進入下一間。

glVertex2f:

四點vertex(x,y,z,w)乘ModelView\_Matrix再乘Projection\_Matrix會得到新的(x,y,z,w)。

四點個別填入glVertex2f(x/w, y/w)中即可畫出。