1. **问题提出**

设计程序实现下面的L-system

* Characters：“A”, “B”, “[” “]”, “(”, “)”
* Grammar rule
  + A→AA; B→A[B]AA(B)
  + 2 Iterations
    - B🡪 A[B]AA(B) 🡪 AA[A[B]AA(B)]AAAA(A[B]AA(B))
* Character explanation(字符解释)：
  + “A” stands for a stem; “B” is a branch
  + “[]” : the branch turns left with 45°
  + “()” : turn right, 45°
* Could you please give more flexible explanation such that the above scheme can generate 3D tree models (修改相关参数生成3D的树干模型)

1. 算法描述
2. 构造字符串

一开始字符串为”B”,树每生长一次，就遍历字符串，将B替换为A[B]AA(B)，将A替换为AA。

详细见regularStr函数。

1. 按照字符串生成对应的图形（详细见drawTree函数）

需要先对字符的意义进行解析，

A和B都表示在原来的方向上步进一个单位，这里需要定义步进起点和方向，我用弧度α来表示方向，y2=y1+sinα，x2=x1+cosα，通过画出线段（x1,y1）--(x2,y2)，就可以画出所需要的直线。使用弧度有一个好处，就是可以避开斜率不存在情况的讨论。

遇到[，则α加45°，画了[内的线段后，必须回退到这一段的起点，并且改回原来的方向，这里用递归实现，可以比较方便地返回原来的起点，遇到]的时候，返回上一层，并且将α减45°，

遇到(同样处理，不过是α先减45°。

实现思路：

逐个遍历s字符串的字符，遇到A或B，画出（x1,y1）--(x2,y2)，

遇到[，α加45，进入枝的内部画，直到遇到]返回，可以保证回到原来的起点。α减45。

遇到(，α减45，进入枝的内部画，直到遇到)返回，α加45°，

继续扫描下一个字符，重复以上动作。

c）如果要画3D树，则需要在x,y,z空间内做旋转，需要使用球坐标系旋转，应该需要两个弧度调整旋转角度。

1. **功能简介**

先输入树的生长次数，调用drawTree绘制树。

除了题目要求的动作之外，添加了鼠标监听和键盘监听，

鼠标左击，减小投影区域大小，保持窗口大小不变，相当于放大图形，

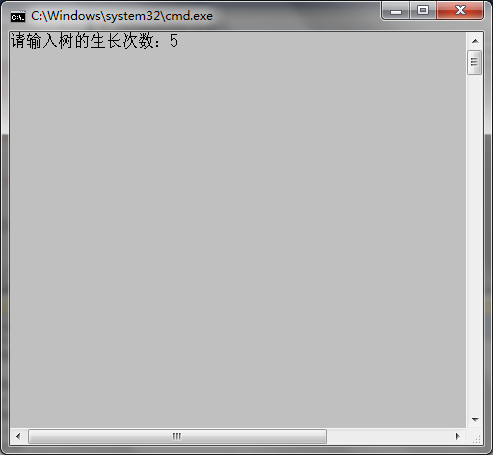
鼠标右击，缩小图形。

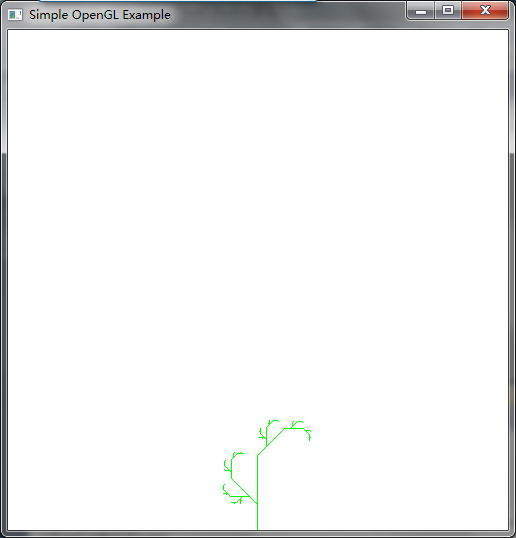
键盘按下Q键，增大树的生长次数，树生长。

按下w键，减小书的生长次数，树退化。

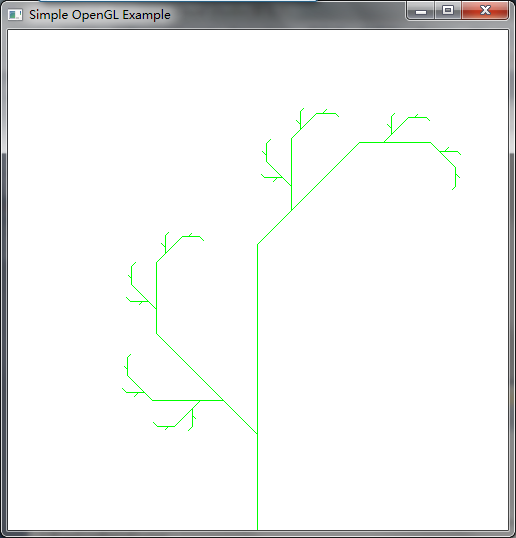
这样可以动态地看到树的生长情况，并且通过缩放观察到整个图形。

1. **实验结果**

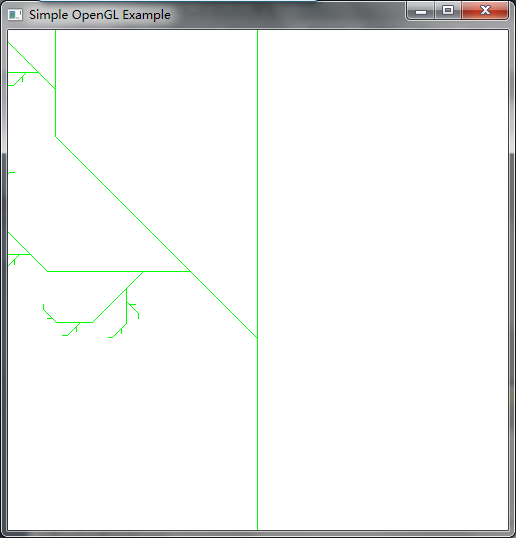




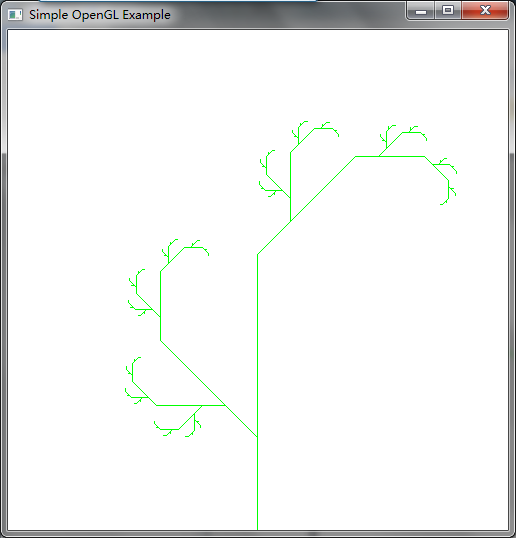
左击放大



按Q生长



右击缩小：



按w退化：

