C++和OpenGL实现吃豆人游戏

项目介绍：

通过 C++ 和 openGL 库来实现了一个吃豆人的小游戏。主要的课程内容涉及到 C++ 相关的序列容器 vector 、deque 和迭代器，以及 openGL 相关的窗口初始化与绘图功能

项目环境：

Microsoft Visual Studio 2010 ，安装openGL

项目分析：

1. 模块分解

在开始编写程序之前我们需要利用 C++ 的思想把实际问题转换成一个个对象。说到游戏应该就能想到角色、地图和怪物以及操作命令等。我们把这些元素细化就可以得到游戏设计模块：

* 角色设计
* 地图设计
* 怪物设计
* 食物设计
* 操作设计
* 界面设计

1. 开发说明：c++：编程语言

Microsoft Visual Studio 2010：开发环境

OpenGL:第三方库

1. 提前准备：我们这次的项目命名为 Pacman。进入这个文件夹创建三个子文件夹 inlcude(存放头文件)、src(存放源代码)和 bin(存放可执行文件)。

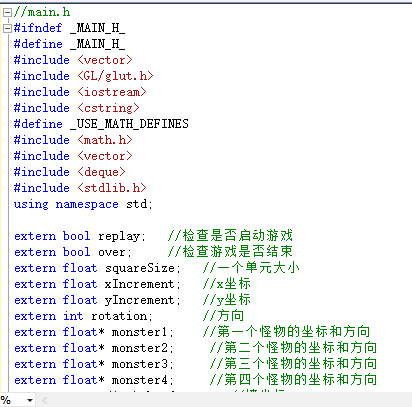
include 目录下

* + control.h
  + createpacman.h
  + food.h
  + gameover.h
  + gameresult.h
  + gamestart.h
  + init.h
  + laberynth.h
  + main.h
  + monster.h

src 目录下（相对应）

* + control.cpp
  + createpacman.cpp
  + food.cpp
  + gameover.cpp
  + gameresult.cpp
  + gamestart.cpp
  + init.cpp
  + labernth.cpp
  + Pacman.cpp
  + monster.cpp

1. 首先在 include 创建 main.h，用来存放头文件和一些全局变量，全局变量的定义是方便所有的模块访问并反映游戏的状态。全局变量定义后，在主函数中进行赋值，所有模块就都能访问了。



5.在主函数中对全局变量进行赋值，然后初始化一个窗口。在这里将使用到 openGL 的初始化接口。具体包括：

void glutInit(int\*argc,char\*\*argv)；初始化

void glutInitDisplayMode(unsighed int mode)；定义显示方式 mode：是一个 GLUT 库里预定义的可能的布尔组合，使用 mode 去指定颜色模式，数量和缓冲区类型。

void glutInitWindowSize(int width,int height);设置窗口大小 width：窗口宽度 height：窗口高度

void glutInitWindowPositon(int x,int y);确定窗口位置（默认左上角） x:横坐标 y:纵坐标

Int glutCreateWindow(char\* title);设置窗口的标题 title:标题内容

void glutDisplayFunc(void(\*func)(void);注册当前窗口的显示回调函数 void (\*func)(void)：回调函数名称，在这里我们用的是 display

void glutReshapeFunc(void(\*func)(int width,int height));重新设置窗口 void(\*func)(int width,int height):回调函数名称,在这里我们用的是 reshape

void glutIdleFunc(void(\*func)(void));调用渲染函数 void(\*func)(void):回调函数名称,系统空闲时调用，在这里我们用的是 display

void glutKeyboardFunc(void(\*func)(unsigned char key,int x,int y));处理按键事件

void glutKeyboardUpFunc(void (\*func)(unsigned char key,int x,int y));处理松开按键事件

void glutMainLoop(void);循环执行

6.