项目概要

/\*贪吃蛇小游戏\*/

1. 基本功能要求：
2. 积分功能
3. 暂停、继续功能
4. 加速、减速功能
5. 背景音效播放
6. 游戏存档功能

(二)基本功能具体实现方法：

蛇分为一个一个身体构成，利用vector容器存放每一个蛇身节点，抽象为蛇类由身体类继承而来，身体类有位置坐标、身体颜色等私有属性（可按需添加）。蛇类有存放蛇身体的容器、保存最后一个节点、保存积分等私有属性，还有移动方向、临时方向等公有属性。

相应的还有成员方法。

**身体类**

**{**

**方法：**

**1、绘制身体**

**2、身体移动**

**3、碰撞检测（检测身体对象之间有没有相互碰撞/位置坐标有没有重合）**

**4、获取身体节点（x,y）**

**5、设置身体节点坐标（在蛇身移动的时候，用来获取节点坐标，一节一节往前移动就行）**

**属性：**

**1、x坐标**

**2、y坐标**

**}**

**蛇类 继承于身体类（可以选择不同的蛇）**

**{**

**方法：**

1. **绘制节点（绘制身体类对象，调用身体类的绘制函数）**
2. **移动节点/身体，采用节点前移的方法：从末尾节点开始，循环获取当前节点的前一个节点的位置坐标等信息，赋值给当前节点。所以只需要移动头节点就行（利用身体类里面的身体移动函数结合方向信息，改变头节点的位置）**
3. **碰撞检测，同样利用头节点的位置，与食物位置是否相等，来判断蛇是否吃到食物。**
4. **由于节点信息，积分是私有成员，就需要定义获取节点信息，积分等的工具函数，方便在类外调用信息。**

**属性：**

1. **节点容器（存放多个身体类）**
2. **积分**
3. **末尾节点**

**}**

**食物类 继承于身体类**

**{**

**方法：**

**1、绘制食物（本质上也是身体）**

**具体实现：利用随机刷出的积分值，找到对应的颜色节点（食物）再绘制出来。**

**2、改变食物位置（每次吃到食物后，都会随机刷新食物的位置，同时生成随机积分）**

**3、利用cjson库解析游戏存档，获取food对应的键和键值。更新需要绘制的食物节点。**

**4、获取食物节点的x，y坐标和颜色函数。（因为食物类继承于身体类，里面不需要在重新定义x,y,color变量了）。**

**属性：**

1. **map容器（以积分为键，颜色/食物类型作为键值创建一个map容器）**
2. **每个食物所代表的积分**

**}**

**场景类**

**{**

**方法：**

1. **运行函数：包括绘制食物，绘制蛇身，移动蛇身；碰撞判断（吃到身体/食物，碰到墙壁）；**
2. **显示积分等信息。（用到了字体库，字体可以悬浮显示在游戏界面上）**
3. **创建json格式的游戏存档文件。获取当前场景下所有的snake节点信息，和食物节点信息，以json格式存储，方便利用cjson库解析。**
4. **获取当前场景类的私有成员的函数**

**属性：**

1. **蛇类对象**
2. **食物对象**
3. **存放文字缓冲区buf**

**}**

(三)触摸屏操作，lcd图片显示操作

触摸屏类

{

方法：

1. 构造函数（打开触摸屏设备文件）
2. 析构函数（关闭触摸屏设备文件）、接合游戏运行界面线程
3. 游戏界面运行函数（由一个子线程执行）；
4. 读取触摸屏触点运行函数run(),根据触摸不同的区域执行不同的功能函数。

}