**贪吃蛇大作战项目用例描述说明书**

**作者：**162052109魏学良 162052113任晓迪

**文档变更记录**

| 日期 | 版本号 | 修订内容 | 修订 | 审核 |
| --- | --- | --- | --- | --- |
| 2018-10-08 | V1.0 | 引言项目概述 | 魏学良 |  |
| 2018-10-08 | V1.1 | 项目用例描述 | 任晓迪 |  |
| 2018-10-15 | V1.2 | 3.1-3.4 | 魏学良 |  |
| 2018-10-16 | V1.2 | 3.5-3.8 | 任晓迪 |  |

**1.引言**

在c#的学习过程中，运用所学内容按照常规项目的步骤流程完成贪吃蛇的游戏，希望通过本次项目对游戏开发流程有一定认识，对所学知识加以运用，增强团队合作的意识和理解。参考自CSDN论坛。

**2.项目概述**

在c++和java语言的基础上，对c#进行学习并尝试完成一个小游戏的制作（贪吃蛇），与同学交流讨论后形成一个两人小团体来共同完成此项目。当然随着微信小游戏的爆红，小游戏的开发也越发吸引人们去探索，因此此项目不仅是对知识的实践更是顺应时代步伐的契机，我相信这对当下的我们意义重大。在此过程中计划通过git平台实现代码分工，组合和版本的迭代，做到精益求精，注重过程的体验。

**3.** **项目用例描述**

**3.1 初始界面**

**3.1.1正常处理**

**【用例名称】**

初始界面

**【场景】**

Who：玩家

Where：游戏启动后的第一个场景

When：游戏启动后

**【用例描述】**

1、 游戏开始运行后显示欢迎界面，并提示游戏正在加载

2、 显示游戏主界面（地图）、当前的关卡数、获得的分数，以及游戏操作菜单（开始游戏、暂停游戏）

3、主界面中央显示长度为3的蛇，蛇头朝右

**【用例价值】**

游戏初始化完成后，实现了游戏初始化的绘制，玩家可以获得游戏的初始信息，并且为游戏的开始提供初始环境。

**【约束和限制】**

1、 游戏主界面地图通过设置合理的坐标，来对蛇的运动范围及位置进行控制；

2、 舌头和蛇身体通过不同的字符表示。

**3.1.2异常处理**

无

**3.1.3替代处理**

无

**3.2 蛇的运动**

**3.2.1正常处理**

**【用例名称】**

蛇的运动

**【场景】**

Who：蛇

Where：欢迎界面显示之后游戏主界面

When：游戏初始化完成后，游戏运行整个过程中，蛇死亡之前，游戏结束前

**【用例描述】**

1、蛇头按照当前场朝向，先前移动一个坐标单位（若干个像素组成一个单位），

2、从蛇头以后的第一节身体移动到蛇头之前的位置，从第二节身体开始，每一节身体移动到前面一节身体的位置。

**【用例价值】**

实现蛇的整体移动，保证游戏正常运行。

**【约束和限制】**

1、 蛇是由两部分组成（蛇头，蛇身）；

2、 蛇只能在规定界面内运动且不可接触既定的边界

**3.2.2异常处理**

无

**3.2.3替代处理**

无

**3.3 蛇头的方向控制**

**3.3.1正常处理**

**【用例名称】**

蛇头的方向控制

**【场景】**

Who：蛇头

Where：欢迎界面显示之后游戏主界面

When：游戏初始化完成后，游戏运行整个过程中，蛇死亡之前，游戏结束前

**【用例描述】**

1、 蛇头的方向受键盘指令的支配，比如蛇想要向左移动，此时蛇头的坐标（x，y）中的x会准备变化（减小）

**【用例价值】**

实现对蛇运动方向的控制，提高了游戏的操作性

**【约束和限制】**

可通过键盘的WASD(W-朝上、A-朝左、S-朝下、D-朝右)对蛇头的方向进行控制

**3.3.2异常处理**

添加1功能多指令的方法,当W(A、S、D)键故障后，可以通过小键盘的上下左右来控制

**3.3.3替代处理**

**3.4 蛇吃食物**

**3.4.1正常处理**

**【用例名称】**

**蛇吃食物**

**【场景】**

Who：蛇、食物

Where：游戏主界面地图内

When：游戏初始化完成后，游戏运行整个过程中，蛇死亡之前，游戏结束前

**【用例描述】**

1、 首先食物同样是由若干个像素点组成的单位（坐标），但开始他们会随机出现在地图的某个位置，设定一次出现一个食物，被吃掉后，立即刷新出一个新的食物。

2、蛇头的坐标如果和食物的坐标相同，则表示蛇吃到了食物，蛇的长度加1。

**【用例价值】**

通过蛇吃食物来进行游戏进程，完成关卡通过条件，实现游戏主体玩法。

**【约束和限制】**

1. 蛇吃到食物后除了长度增加1，分数也会累加10
2. 食物随机刷新但不能出现在界面的四角

**3.4.2异常处理**

无

**3.4.3替代处理**

无

**3.5 关卡控制**

**3.5.1正常处理**

**【用例名称】**

关卡控制

**【场景】**

Who：关卡

Where：游戏主界面

When：累计的积分达到既定值时

**【用例描述】**

1、游戏开始后累计分数达到指定值后弹出指示界面，提示用户是否进入下一关

2、设置多个关卡，关卡间的区别是对于积分的要求，随着关卡数的递增，过关时的积分数值也在递增

3、进入下一关时，蛇身和蛇头恢复初始长度，

**【用例价值】**

通过设置关卡逐步提高游戏难度，增加游戏的可玩性

**【约束和限制】**

积分足够时游戏自动暂停提示是否进入下一关

**3.5.2异常处理**

无

**3.5.3替代处理**

无

**3.6 死亡控制**

**3.6.1正常处理**

**【用例名称】**

死亡控制

**【场景】**

Who：蛇

Where：游戏主界面地图内

When：蛇头撞到游戏界面边框或蛇身时

**【用例描述】**

1. 当蛇头坐标与边界重合时，弹出游戏结束的提示框，关闭后返回主界面
2. 当蛇头与任意蛇身坐标重合时，弹出游戏结束的提示框，关闭后返回主界面

**【用例价值】**

让游戏的规则束缚更加合理，具有了输赢的属性，增加可玩性

**【约束和限制】**

**3.6.2异常处理**

无

**3.6.3替代处理**

无

**3.7 暂停控制**

**3.7.1正常处理**

**【用例名称】**

暂停

**【场景】**

Who：蛇

Where：游戏主界面地图内

When：按下空格键

**【用例描述】**

1、按下空格时，蛇头停止运动，并弹出游戏暂停的提示框，此时可以选择返回主界面或继续游戏。

2、选择继续游戏时可使蛇头继续运动。

**【用例价值】**

可以使玩家在游戏期间停止游戏，方便游戏管理

**【约束和限制】**

只有在游戏进行中才能暂停游戏

**3.7.2异常处理**

若空格键失效，按下P键游戏停止

**3.7.3替代处理**

按下P键游戏停止

**3.8 退出控制**

**3.8.1正常处理**

**【用例名称】**

退出控制

**【场景】**

Who：蛇

Where：游戏主界面

When：按下退出游戏键

**【用例描述】**

在游戏主界面按下退出游戏时，游戏界面关闭

**【用例价值】**

通过此操作结束游戏

**【约束和限制】**

按下退出游戏，游戏界面关闭

**3.8.2异常处理**

右上角关闭键

**3.8.3替代处理**

右上角关闭键