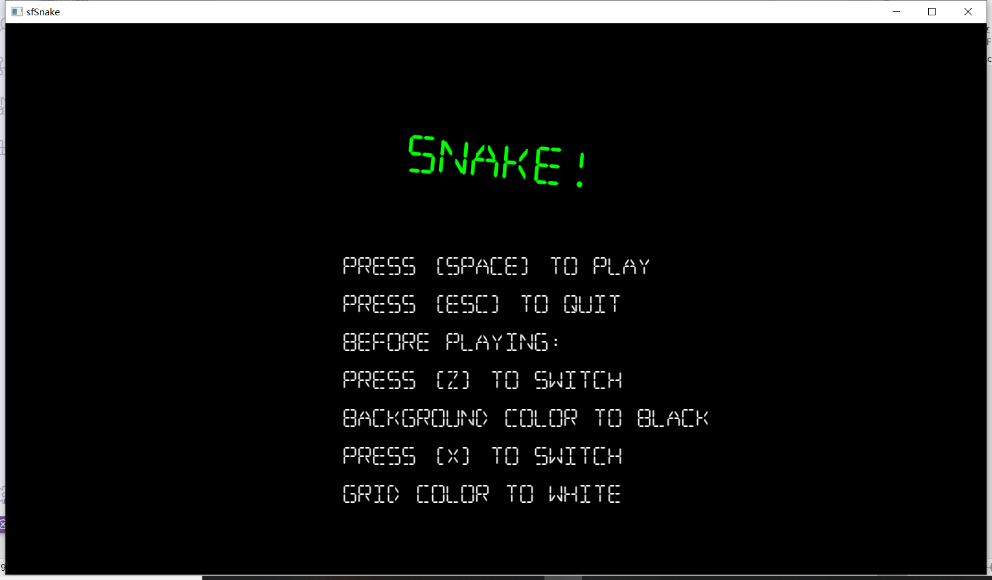
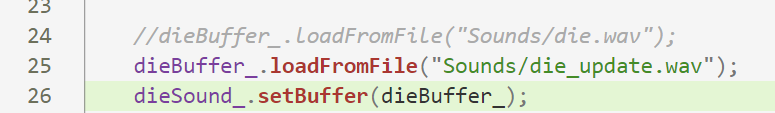
**贪吃蛇游戏设计说明**

梁之扬 19307130184 2019级技术科学试验班

1. **贪吃蛇的控制方法：**
2. 在要求以外，增加了利用 W S A D 四个方向键来控制贪吃蛇的移动，符合笔记本玩家的要求（比如我的笔记本键盘不合理 上下左右箭头的四个按键比较小）
3. 鼠标的反馈均在window 内，故可以放大游玩游戏而不会导致鼠标错位。
4. 为了避免非失误操作造成的自杀行为，游戏设定了贪吃蛇只能接受以运动方向为轴，左右各135度的转向，若鼠标或键盘给的指令在这270度范围之外，则贪吃蛇不会做出回应。具体见Snake.cpp第43行至第64行。
5. **整体界面的修改：**
6. 在紧张而刺激的游戏过程中，玩家的注意力主要在贪吃蛇上，不会考虑修改界面的问题，故我将修改界面的时机放在了主界面中，利用Z键修改背景颜色，利用X键修改网格颜色，而且有实时的反馈，让玩家知道进入游戏后会是什么情况。
7. 修改了结束界面重开时回到的界面，原本是回到游戏中界面，现在是回到游戏前的界面，方便玩家更改背景和网格颜色。
8. 通过更改边界判定的方式，修复了贪吃蛇在窗口边缘时可能散架的问题。
9. 稍微放大了一下原来的窗口，使得游戏更加清楚。
10. **音乐文件的修改和修复**
    1. 修复了损坏的pickup音效，改成了sfml库支持的wav格式文件，使得贪吃蛇吃水果时音效正常。
    2. 修改了背景音乐和死亡音效，死亡音效有些恶搞的意味，会暂停游戏并播放约15s，若需要更改回来，可以在Snake.cpp中将25行替换为被注释掉的24行。

同样的，背景音乐可以在Game.cpp中将22行替换为被注释掉的21行。

1. **杂项**
   1. 通过随机生成水果的代码修复了水果总是生成在同一行的问题，并且通过给水果增加品红色边框来避免与背景颜色相同的水果无法被玩家分辨出来。
   2. 通过修改贪吃蛇自身碰撞代码避免原代码在鼠标操控时导致的贪吃蛇意外死亡的问题，详见Snake.cpp 112-126行。
   3. 蛇身会随着贪吃蛇运动方向改变而改变朝向，使得整体看上去更自然，详见SnakeNode.cpp 43-68行。
   4. 游戏性能方面还算过关，大概需要蛇占满半个屏幕才会导致程序进入无法响应的状态，而占满半个屏幕需要的时间已经可以让人玩腻了（除非他就是来听BGM的）。
2. **代码书写过程当中的一些参考**
   1. 代码是独立完成的，过程中也与别人讨论了很多次，其中有两处是杜雨轩同学给了我很大帮助，而我几乎没有想到什么好的解决办法，故在代码中注释出来。
   2. 有浏览Google、CSDN、StackOverflow、GitHub上的许多参考资料，但最终并没有太大帮助，最后是浏览sfml本地文件中的.hpp文件获得的函数书写帮助。
   3. 感谢源代码的作者，也感谢王雪平老师和助教老师一学期以来的工作，让我们学习C++的知识，也获得了实际的写代码的技能。