高级语言程序设计

实验报告

南开大学 工科试验班

姓名 蔡佳良

学号 2213117

班级 3430

2023年5月11日

[高级语言程序设计大作业实验报告 3](#_Toc134717770)

[一. 作业题目 3](#_Toc134717771)

[二. 开发软件 3](#_Toc134717772)

[三. 课题要求 3](#_Toc134717773)

[1. 面向对象 3](#_Toc134717774)

[2. 单元测试 3](#_Toc134717775)

[3. 模型 3](#_Toc134717776)

[4. 验证 3](#_Toc134717777)

[四. 主要流程 3](#_Toc134717778)

[1. 整体流程 3](#_Toc134717779)

[2. 算法公式 4](#_Toc134717780)

[（1） 输出图像函数（graphics.h） 4](#_Toc134717781)

[（2） 图像处理函数（graphics.h使绘图更为平滑，屏幕不会出现闪烁） 4](#_Toc134717782)

[（3） 获取按键信息函数 4](#_Toc134717783)

[（4） 暂挂函数 4](#_Toc134717784)

[（5） 初始化函数 （graphics.h） 4](#_Toc134717785)

[（6） 播放音乐函数（mmsystem.h） 4](#_Toc134717786)

[3. 测试运行 4](#_Toc134717787)

[五. 测试 5](#_Toc134717788)

[1. 静止 5](#_Toc134717789)

[2. 移动 5](#_Toc134717790)

[3. 同时移动 5](#_Toc134717791)

[4. 跳跃 6](#_Toc134717792)

[5. 普攻 6](#_Toc134717793)

[6. 远攻 6](#_Toc134717794)

[7. 受击 7](#_Toc134717795)

[8. 击倒 7](#_Toc134717796)

[9. 结算 7](#_Toc134717797)

[六. 收获 8](#_Toc134717798)

高级语言程序设计大作业实验报告

1. 作业题目

简易版死神VS火影游戏C++复刻。

1. 开发软件

Visual Studio 2022

1. 课题要求
2. 面向对象
3. 单元测试
4. 模型
5. 验证
6. 主要流程
7. 整体流程

实现思路：

定义角色结构：（用于记录角色的位置、生命值、体积等）

struct role

定义角色类：

class Role

成员方法为控制角色移动的函数和判断角色是否移动的函数：

void PrintStill();

void PrintRun();

int IsRunA(role&);

……

1. 算法公式
2. 输出图像函数（graphics.h）

void putimage(int dstX, int dstY, const IMAGE \*pSrcImg, DWORD dwRop = SRCCOPY);

//在（x,y）处输出图像。

void putimage(int dstX, int dstY, int dstWidth, int dstHeight, const IMAGE \*pSrcImg, int srcX, int srcY, DWORD dwRop = SRCCOPY);

//在（x,y）处输出指定大小的图像

1. 图像处理函数（graphics.h使绘图更为平滑，屏幕不会出现闪烁）

BeginBatchDraw();//开始连续绘图

EndBatchDraw();//停止连续绘图

1. 获取按键信息函数

GetAsyncKeyState()//按下对应按键则返回值

1. 暂挂函数

Sleep（）//执行挂起一段时间，也就是等待一段时间在继续执行

1. 初始化函数 （graphics.h）

Initgraph（）//初始化一个窗口

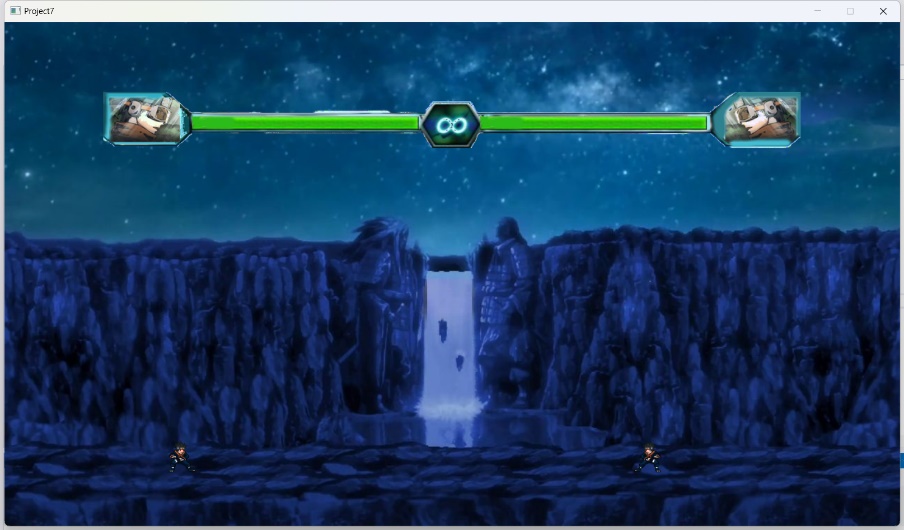
1. 播放音乐函数（mmsystem.h）

mciSendString（）

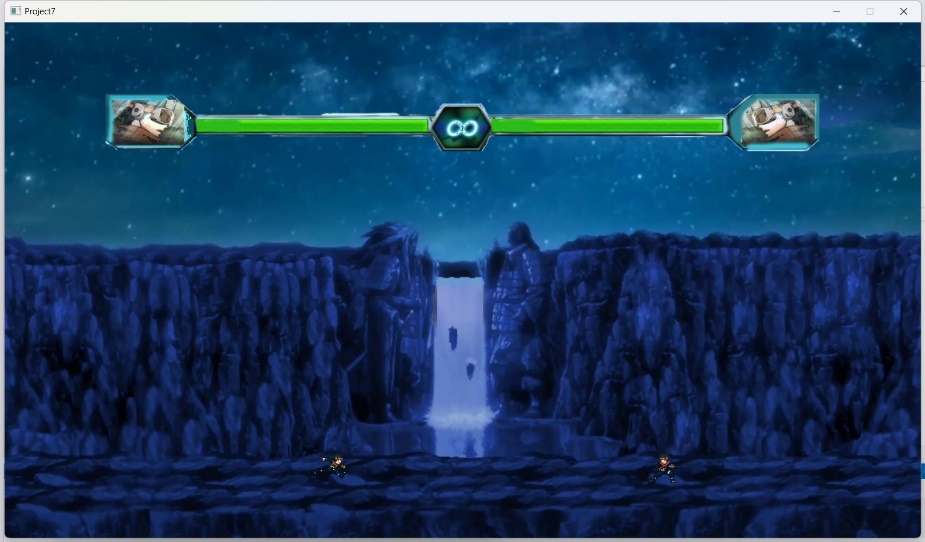
1. 测试运行

使用J,K,A,D,U,→,←控制角色移动和攻击，测试角色是否能正常运动，角色死亡后能否正常结束。

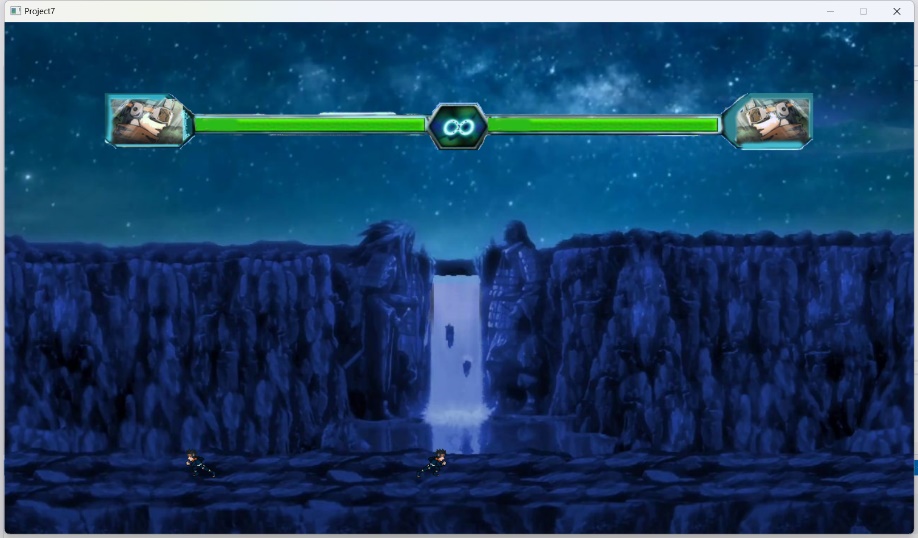
1. 测试
2. 静止



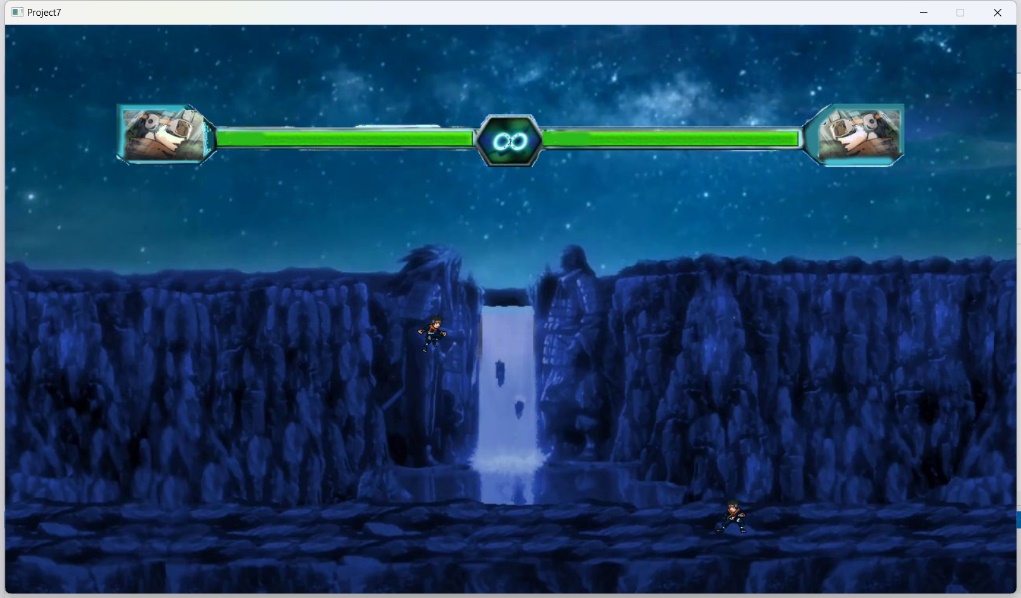
1. 移动



1. 同时移动



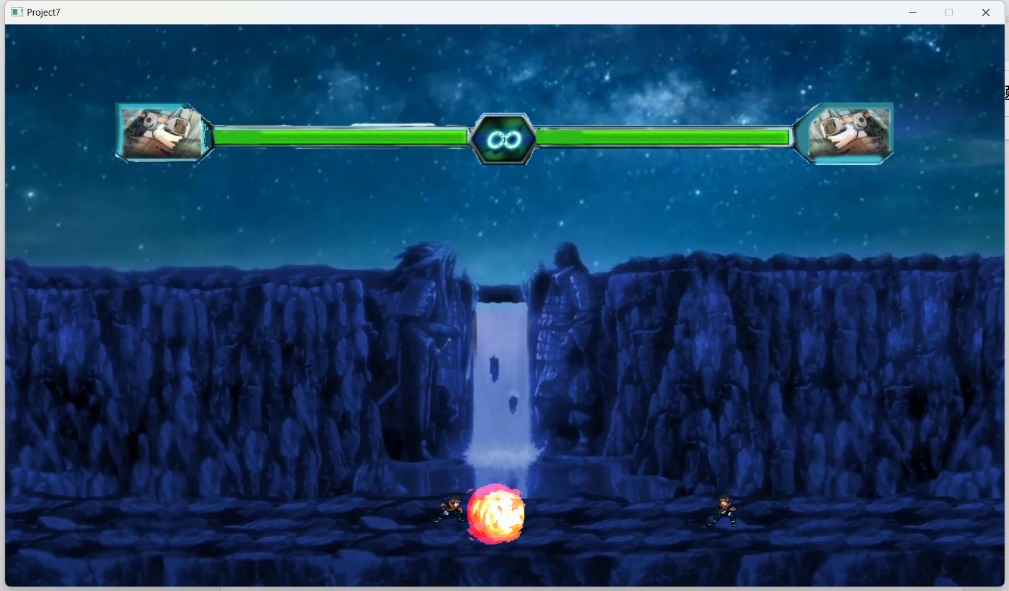
1. 跳跃



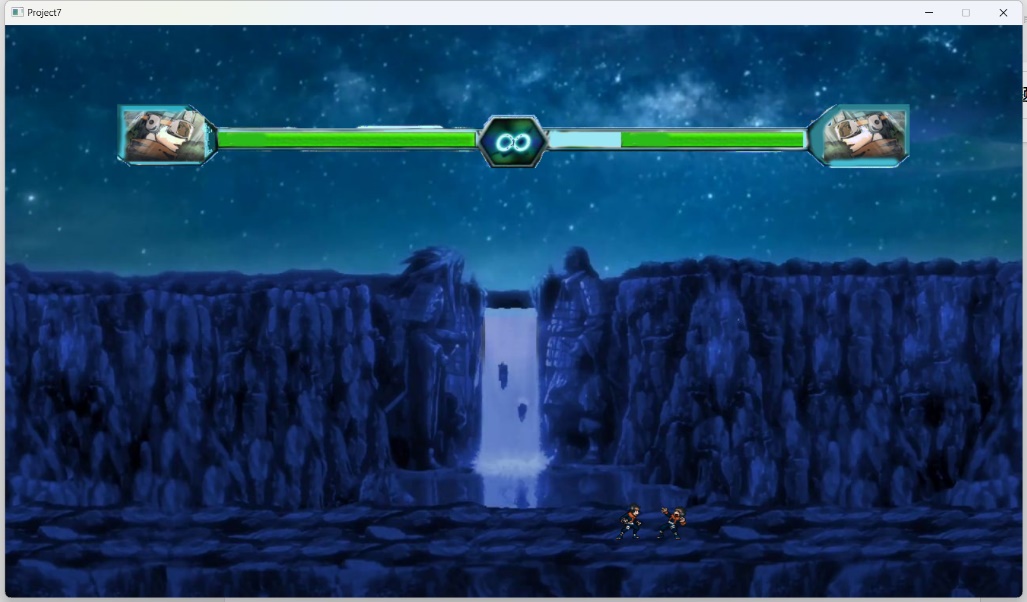
1. 普攻



1. 远攻



1. 受击



1. 击倒



1. 结算



1. 收获
2. 使用GetAsyncKeyState()函数获取按键信息而不是\_getch()函数：

GetAsyncKeyState()函数所获取为硬件信息反映的为即刻按键状态；而\_getch()函数获取的为输入函数，当长按或者按连续按多个键时往往会积累多个输入导致程序获取的按键与实际按键不统一。

1. 实现两个人物的状态：操作一个人物时需要判断另一个任务的状态并输出