

项目报告

刘真远，何凡，吴宇轩

2025 年 1 月 12 日

1 引言

本项目为 SI100B 课程的最终项目，项目文件已上传到

<https://github.com/liuzhy673/SAO>。

我们制作了一款RPG类型的游戏。玩家初始出生在森林，拥有 1000 金币武装自己，随后可通过传送门进入野外与怪物搏斗。

战斗采用回合制，玩家与怪物轮流攻击，直到一方死亡。游戏中有三种怪物：普通狗头人小怪，狗头人精英，最终boss，每种怪物的属性都不一样。

玩家击败怪物后可以获得金钱。

返回城市可以用金币购买更为强力的物品：商店界面采用鼠标操作，选择自己想购买的升级并点击即可将其收入囊中，分为增强攻击，防御，生命上限，以及复活药水4种药水。本游戏采用一命通关制，如果复活药水为0，一旦死亡，立刻退出游戏。

击败 Boss就能游戏通关，并进入结尾动画，可按空格键跳过。

在运行本程序前，由于不同设备的差异，请将整个打包的zip解压后直接放在C盘，不要再添加文件夹或其他位置，否则程序可能无法正常运行。并且在运行本程序前，请提前下载好cv2，以便完整的游戏体验。

2 项目实施

2.1 场景

2.1.1 场景数量

我们的游戏中一共有三个场景：

- 1、新手村: 玩家可以在此处与 NPC 对话，在商店中用金钱进行交易，见图1；
- 2、金币关: 玩家可以在此处与怪物战斗获取金钱，见图2；
- 3、主线关卡: 进入后开始游戏主线，见图3。

我们的场景由一张大图构成。

2.1.2 大型场景

我们游戏中的每个场景都是大型场景，都有着可以跟随玩家移动而进行视角移动的摄像头。



图 1: 新手村场景



图 2: 金币关场景



图 3: Boss 房场景

镜头移动的原理：每当满足镜头移动的条件时，所有物体（含背景、NPC、玩家）都会获得一个偏移量，该偏移量等于在这一帧玩家试图移动的距离，在渲染时所有物体的位置都会加上这个偏移量的相反数。因此，玩家移动了 $(dx, dy) + (-dx, -dy) = (0, 0)$ ，即在原地不动，其它物体全部移动偏移量的相反数，达到了镜头移动的效果。

2.1.3 互动物品

我们的游戏中有两种互动物品，分别为**宝箱**，**墙壁**，如图 4。



图 4: 互动物品

其中，墙壁起到障碍物的作用，玩家无法跨越障碍物。宝箱则是在地图位置随机生成，捡到后能获得一定数量的金币。

2.2 角色

2.2.1 主要角色

在我们的游戏中，玩家可以通过 wasd 四个按键操控主要角色四处移动，通过传送门进入其他场景，通过与其他角色碰撞来触发特殊事件。主要角色的移动通过检测按键按下与抬起这两种事件的方法来实现，当按键被按下，角色就会持续朝指定方向移动，按键抬起后，角色不再朝指定方向移动。按下按键等事件的检测全部在 `pygame` 中的各类 `update` 方法中实现。

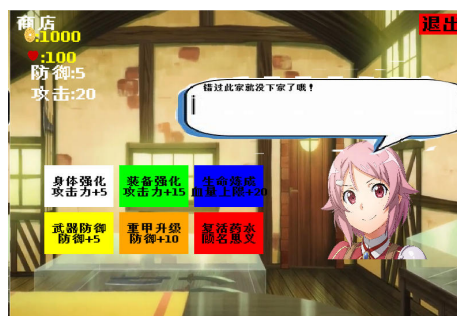
玩家的横向移动与纵向移动被写成了两种方法，在每一帧中，程序会先进行横向移动，判断与障碍物的碰撞，若碰撞则撤回横向移动，再进行纵向移动，判断与障碍物的碰撞，若碰撞则撤回纵向移动。这种分开处理横纵移动的移动处理方式使得玩家在障碍物边缘同时按下两个方向键时不会被卡住。

2.2.2 友好 NPC

在城市场景中存在着两种友好 NPC，一种是对话 NPC，还有一种是商店 NPC，见图 5。



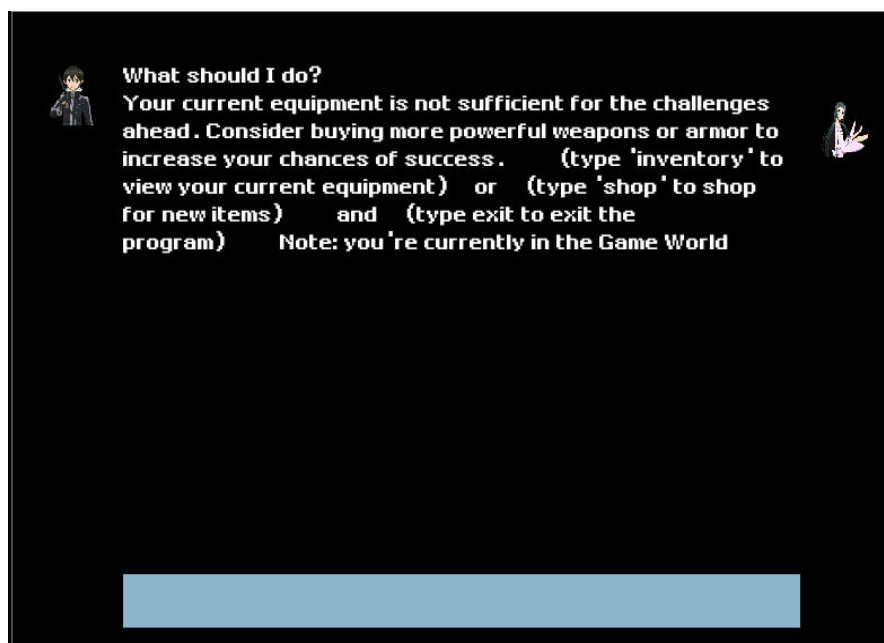
(a) 对话 NPC



(b) 商店 NPC

图 5: 友好 NPC

当角色与对话 NPC 碰撞时，会弹出对话框，此时玩家可以选择按 退出 键退出对话，也可以选择按武器键进入购买。对话NPC在开始会给与游戏提示，告诉玩家游戏步骤，没能遵守游戏步骤的话则直接进入大语言模型环节，会与NPC直接进行对话。我们对大语言模型的回答做了一定限定，但不能确保他不会自由发挥一点。大语言模型中默认蓝灰色为可输入状态，深蓝色为不可输入状态，每次对话后都会清屏。我们商店NPC当钱不足时不会跳出商品的详细信息。



2.2.3 简单敌人

金币关场景中存在简单敌人：狗头人护卫。见图6。



图 6: 简单敌人

2.2.4 特殊敌人

主线关卡中存在着两种特殊敌人：狗头人精英以及最终幕后boss：见图7。

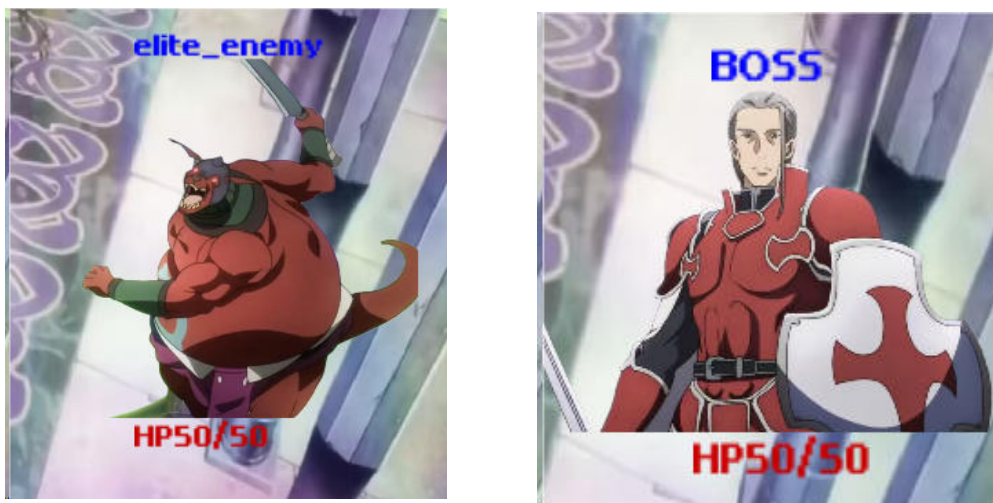


图 7: 狗头人精英与最终boss 茅场晶彦

2.3 游戏机制

2.3.1 核心机制

玩家通过操作主要角色移动来与要互动的角色碰撞，碰撞发生后即会触发相应事件。战斗采用回合制，玩家与怪物交替发起攻击。玩家需要反复击败怪物以获取金钱对自己进行升级，而怪物也会在每次玩家进入主线关卡时变得更强大，玩家的最终目标是变得足够强大去挑战。在挑战时玩家拥有三个技能：普通攻击，火球攻击以及恢复。火球攻击的特点是范围攻击。并且穿透防御，但伤害相比普通攻击有所削减；普通攻击即表面的攻击数值，但实际伤害需要减掉怪物本身的防御，随着等级增加，怪物的防御，攻击也会越来越高，并且精英怪与最终boss有不同阶段；恢复血量为攻击的两倍，所以点技能时请慎重考虑。

2.3.2 碰撞系统

程序通过 pygame 内置的碰撞检测方法检测玩家与实体间的碰撞。全部的碰撞检测全部由我们经过坐标的反复测试，选定最好的点位作为碰撞点来进行。

当玩家与障碍物发生碰撞时，当坐标到达一定位置则按键判定无效，已达到碰撞效果。

当玩家与宝箱、NPC、商店、传送门等实体碰撞时，碰撞检测方法会向事件队列发送相对应的事件，在事件队列中被相继处理。

2.3.3 资源系统

玩家击败怪物就可获得金币，怪物的等级越高，玩家获得的金币就越多。金币可以用来在商店中购买升级，提高玩家的战斗力。金币的用途在商店中以物体的形式呈现。初始金币看似很多，但由于我们采用一命通关制，想直接一命通关的话道具点一定要点好。虽然没有设定刷金币关卡的次数，但依然希望大家能够只进一次金币关卡就能够一命通关（真的没这么简单）。

2.4 游戏性

2.4.1 菜单

我们的游戏有三种菜单，主菜单、失败界面与胜利界面。

进入游戏与游戏重新开始时会显示主菜单，此时玩家需要按下 **开始游戏** 键以进入游戏。

2.4.2 BGM

我们的游戏在战斗时会播放**慷慨激昂**的BGM

2.5 代码

仓库中 **main** 为游戏文件。用编辑器打开 后运行 **Main.py** 即可进入游戏。

程序采用面向对象编程的编程方式，代码中功能以及注意点全都添加了注释。

下面罗列程序中的各个文件的相应作用：

- 1、**main.py** 该文件仅作为**游戏主体，最简版，负责运行游戏。**
- 2、**environment.py** 该文件是进行游戏时的主程序，同时由于后期各种补丁不作为最终**main**。
- 3、**setting.py** 该文件定义了三场战斗的**详细代码**。
- 4、**storecode.py** 该文件定义了商店系统。
- 5、**ceshi.py** 该文件**展示**我们的大语言模型。
- 6、**coin_number.py** 该文件**展示**我们金币系统的**联络**
- 7、**data_module.py** 该文件**展示**我们的初始数据

