

**问题求解实战结题报告**

**计算机学院**

**2025年6月15日**

**项目名称：幻影迷宫**

**学号：2023141460164**

**姓名：刘畅**

**指导教师：张轶**

**目录**

[一、 游戏概述 4](#_Toc201180295)

[1. 游戏介绍及特色 4](#_Toc201180296)

[2. 开发环境 5](#_Toc201180297)

[3. 游戏工作量 5](#_Toc201180298)

[4. 总体流程 7](#_Toc201180299)

[二、 游戏内容 8](#_Toc201180300)

[1. 开始界面 8](#_Toc201180301)

[2. 说明界面 9](#_Toc201180302)

[3. 游戏界面 9](#_Toc201180303)

[4. 暂停界面 11](#_Toc201180304)

[5. 失败界面 11](#_Toc201180305)

[6. 胜利界面 12](#_Toc201180306)

[7. 主角 12](#_Toc201180307)

[1) 基本运动状态 13](#_Toc201180308)

[2) 攻击能力 13](#_Toc201180309)

[3) 控制方式 14](#_Toc201180310)

[8. 敌人/敌对单位 14](#_Toc201180311)

[1) 奶龙 14](#_Toc201180312)

[2) 幽灵 15](#_Toc201180313)

[3) Boss 16](#_Toc201180314)

[9. 道具 17](#_Toc201180315)

[10. 地图生成和传送机制 19](#_Toc201180316)

[三、 代码结构与核心逻辑 21](#_Toc201180317)

[1. 文件结构 21](#_Toc201180318)

[2. 代码逻辑设计 22](#_Toc201180319)

[1) 状态管理系统 22](#_Toc201180320)

[2) 地图与关卡系统 23](#_Toc201180321)

[3) 角色控制系统 25](#_Toc201180322)

[4) 碰撞检测系统 28](#_Toc201180323)

[5) 事件系统 31](#_Toc201180324)

[6) 音效与资源管理 35](#_Toc201180325)

[7) 模块间通信机制 35](#_Toc201180326)

[四、 心得体会 36](#_Toc201180327)

# 游戏概述

## 游戏介绍及特色

《幻影迷宫》（Phantom Maze）是一款2D俯视视角的动作冒险闯关游戏，融合了战斗、解谜与地图探索元素。玩家将在游戏中扮演一只灵巧的蝙蝠，穿梭于六大风格各异的迷宫关卡，解锁出口、收集资源、击败敌人，最终挑战强大的Boss。

每个关卡地图由墙壁、地面、道具与机关等元素构成，玩家需要灵活躲避敌人（如奶龙、鬼魂），利用声波攻击清除障碍，拾取血包恢复生命，通过收集金币强化攻击力或购买大招次数。有些关卡还设置了钥匙机制，玩家需先解锁大门才能进入Boss房。Boss战中，敌人拥有火球轰炸、召唤鬼魂等多种攻击方式，玩家需抓住攻击空隙释放声波或八方向大招精准反击，策略性十足。

本游戏拥有以下几大特色：

1. **多样化关卡设计**：六大普通迷宫加一场Boss战，每关结构、敌人刷新与道具分布均不同，提供持续的新鲜感与挑战；
2. **丰富的成长系统**：金币可用于购买强化（如攻击距离或技能使用次数），蝙蝠道具可提升声波范围，玩家可自由规划成长路线；
3. **真实战斗体验**：怪物AI具备巡逻与追踪机制，Boss具备多阶段攻击形态，爆炸、冷却、击退等机制均细致设计，提升沉浸与观赏感；
4. **完整UI与交互设计**：包括主菜单、暂停、胜利/失败界面、技能释放提示等，均配有独立UI界面与音效；
5. **简洁流畅的操作体验**：支持WASD或方向键移动、空格释放声波、J键释放大招、数字键购买强化，部分功能支持鼠标点击，适合不同玩家习惯；
6. **结局分明，挑战十足**：击败Boss即可通关，反之生命值归零将重新开始，鼓励反复尝试与策略规划；
7. **暗色美术与背景音乐风格**：整体画风偏神秘压抑，配合富有节奏感的音乐增强氛围，营造沉浸式探险体验。

本项目专为喜欢解谜、探索与战斗融合玩法的玩家设计，关卡难度阶梯式提升，兼顾新手上手体验与高阶挑战深度，是一款富有挑战性和趣味性的原创2D迷宫闯关游戏。

## 开发环境

系统：windows 11

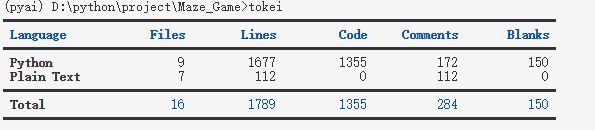
使用软件：

1. Visual Studio Code（开发工具）
2. Adobe Photoshop （素材制作）
3. Python 3.12
4. AI图像生成工具 （素材制作）

编程语言：Python

使用模块：pyzero、pygame等

## 游戏工作量

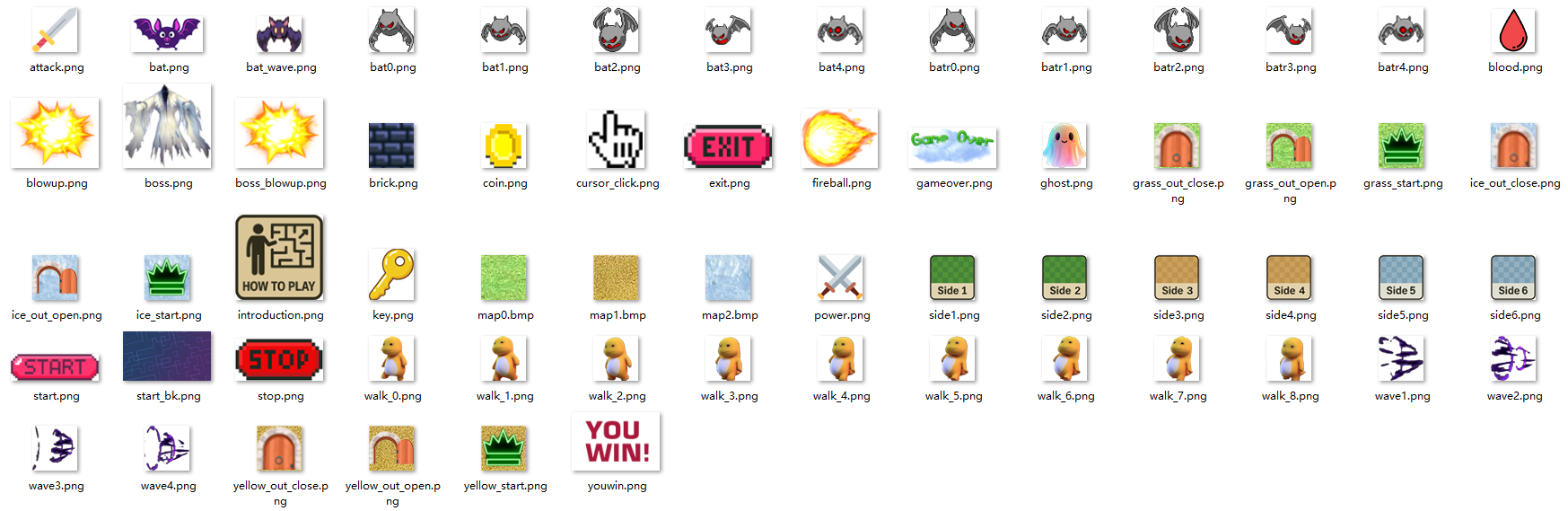


本项目代码部分共包含 16 个文件，其中 Python 源代码文件 9 个，地图数据（Plain Text 格式）文件 7 个。根据代码统计工具 tokei 的结果，项目整体共 **1789 行文本**，其中：

* 1. **核心功能代码**占比最大，约为 **1355 行有效 Python 代码**，涵盖了主界面逻辑、地图读取、角色与敌人行为、碰撞检测、技能实现、UI控制、音效管理、关卡机制等模块，体现了较为完整的 2D 闯关类游戏架构；
  2. **注释部分**达 284 行，表明项目在开发过程中注重代码可读性与后期维护，遵循良好的编程规范；
  3. **空行 150 行**，进一步体现出良好的代码结构与分块设计；
  4. **地图设计部分**通过 7 个文本文件记录了各关卡的字符矩阵结构，便于地图的可视化编辑与程序读取，虽无代码行，但包含完整逻辑元素（如墙体、陷阱、钥匙、出口等）；

游戏中的图像素材量达到了63个，音乐素材量达到了13个。





此外，我在开发过程中使用了git作为版本控制工具，并且托管在Github上。



## 总体流程

玩家打开游戏后，首先进入由美观的背景图片构成的开始界面，屏幕上显示"开始游戏"按钮。玩家可以点击按钮或按特定按键进入游戏介绍界面，该界面详细说明游戏操作和目标。

完成介绍后，游戏加载预设地图，玩家将控制一只蝙蝠角色在迷宫中探索。迷宫中分布着墙壁、道具，以及包括奶龙(MilkDragon)在内的敌人。玩家需要使用方向键或WASD进行移动，通过空格键释放声波攻击来击退敌人。

在地图探索过程中，玩家可能会遇到各种挑战，包括躲避鬼魂(Ghost)的追击。当玩家到达特定位置时，可能会触发Boss战，玩家需要利用声波攻击和机智的走位战胜Boss。Boss会使用多种攻击方式，包括发射火球等。

游戏过程中，玩家可以按下暂停键暂停游戏。如果角色生命值归零，游戏会显示失败界面，玩家可以选择重新开始游戏，重新挑战。

成功通过所有关卡后，玩家将迎来胜利界面，完成整个游戏流程。

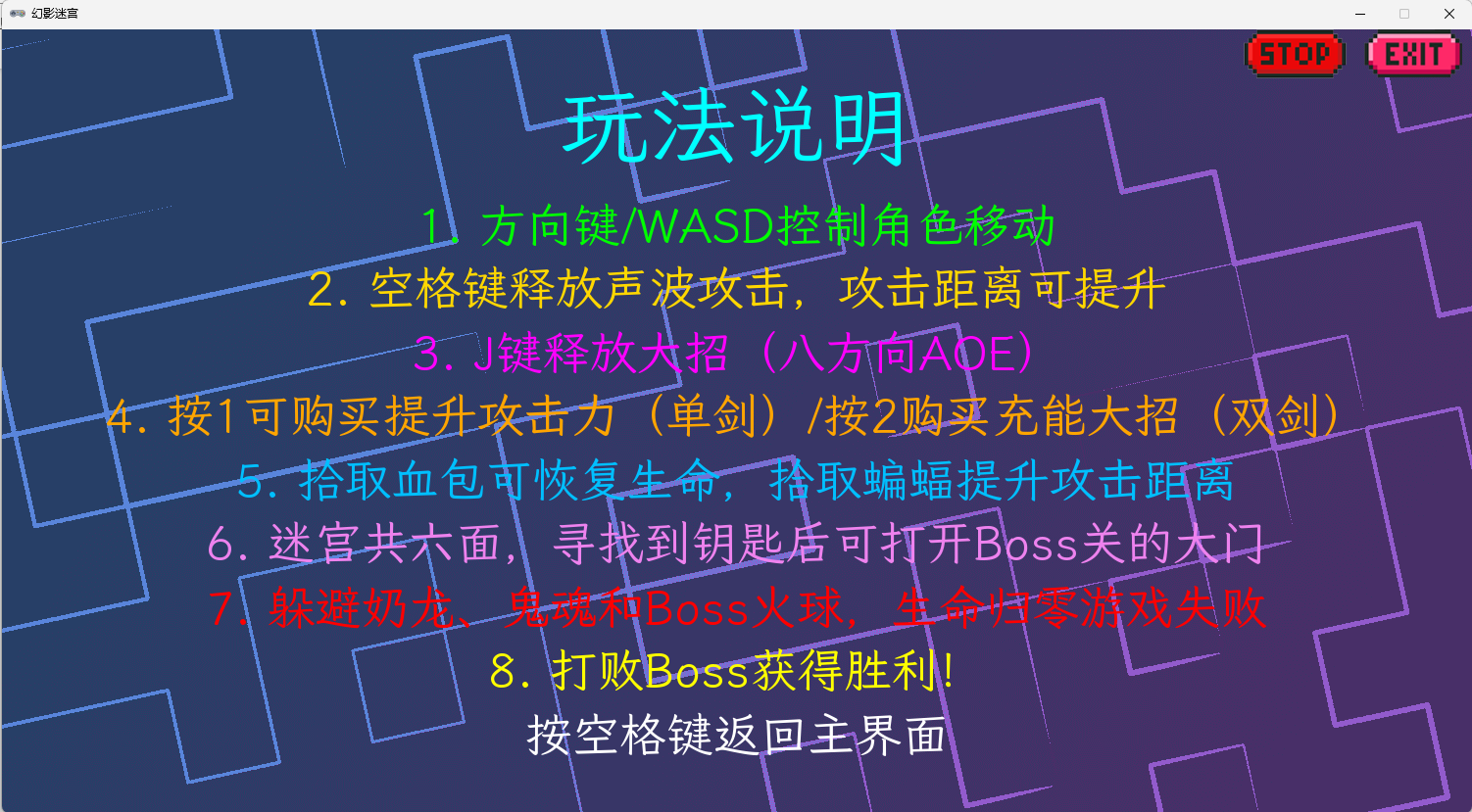
# 游戏内容

## 开始界面



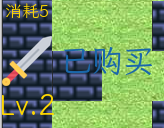
这是进入游戏后的开始界面。

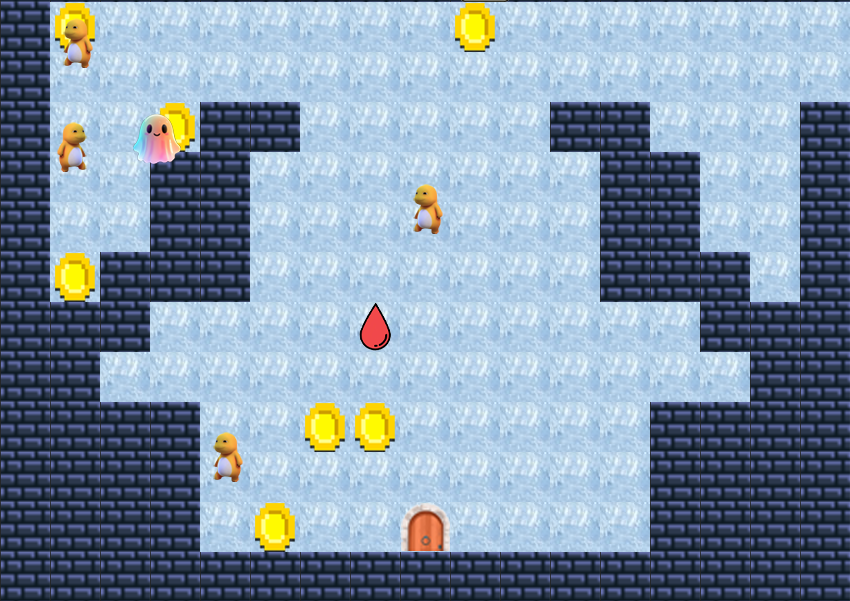
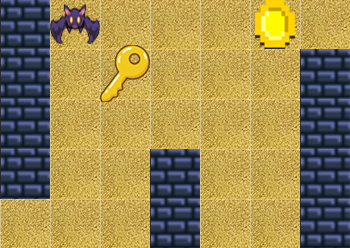
## 说明界面



## 游戏界面



* 在游戏中央区域，是一个由蓝色砖墙和绿色草地构成的迷宫地图，玩家将在其中探索。
* 迷宫中分布有多个金色硬币，玩家可以收集以获得积分或奖励。
* 敌人分布在地图的多个位置，形象为奶龙，会阻碍玩家前进。
* 地图中还存在一个彩色的特殊角色，是特殊敌人监管者。
* 地图中央位置有一个红色水滴形道具，为血包。
* 迷宫的右侧和底部设有门状结构，为当前关卡的传送点。
* 在游戏左上角，会显示当前生命值，即蝙蝠数量。
* 在游戏的顶部中央，有6个标签标识为“Side 1”至“Side 6”，代表多重地图/维度切换功能。
* 游戏右上角包含两个按钮：“STOP”（停止游戏）和“EXIT”（退出游戏）。
* 在游戏左侧状态栏显示“消耗: 5”、“Lv.1”（当前等级）、“消耗: 5”以及“技能”选项，下面显示数值“0”，代表技能数量。
* 游戏左下角会显示“金币 x 0”，表示当前玩家收集的金币数量。
* 购买技能/升级会出现：
* Boss关卡：
* 钥匙以及特殊的Boss关卡传送点：



## 暂停界面

当按下“STOP”时，游戏进入暂停状态，所有角色停止更新。按下空格恢复。



## 失败界面

当生命值为0时，游戏判定为失败。提示按下空格重新开始。



## 胜利界面

打败boss后胜利：



## 主角

蝙蝠角色作为游戏主角，具有以下特性：

### 基本运动状态

蝙蝠角色有两种基本姿态：**向左飞行**和**向右飞行**。

* **飞行动画**：每个方向都有独立的5帧动画序列
* 向左飞行使用 bat0 到 bat4 动画帧
* 向右飞行使用 batr0 到 batr4 动画帧
* **动画切换**：
  + 移动时，每5帧自动切换一次动画状态，形成流畅的飞行效果
  + 静止时，保持当前方向的动画帧，但不再循环播放

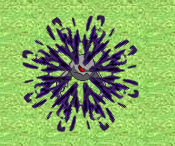
### 攻击能力

蝙蝠具有两种声波攻击模式：

1. **单方向声波**：
   * 按空格键触发
   * 沿当前面向方向发射单向声波
   * 声波范围可通过游戏中拾取蝙蝠道具提升



1. **八方向声波大招**：
   * 按J键触发
   * 同时向八个方向发射声波攻击
   * 需要通过消耗金币购买充能才能使用



声波攻击不能穿墙，遇到墙壁会自动消失。

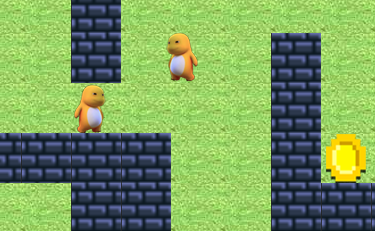
### 控制方式

* 使用方向键或WASD控制移动方向
* 空格键释放单向声波攻击
* J键释放八方向大招

## 敌人/敌对单位

### 奶龙

其原始素材与游戏展示如下： 



奶龙是游戏中的敌对角色，以其可爱的外观和充满活力的动画效果为特点。它拥有完整的9帧行走动画序列（从walk\_0到walk\_8），在移动时会根据方向进行显示调整：向左移动时展示正常图像，向右移动时则通过水平翻转实现朝向变化。当奶龙被击败时，会触发持续15帧的爆炸动画，随后从地图中消失，营造出一定的视觉冲击力。

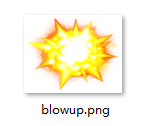
在行为模式上，奶龙采用随机游走的方式进行移动。每隔50帧，它会随机选择一个新方向（上、下、左或右）并持续朝该方向以固定速度前进。在运动过程中，它具备严格的障碍检测机制，确保无法穿越墙体。当遇到障碍物时，会自动改变方向，同时利用更精细的碰撞检测系统避免与墙壁重叠，提升交互的真实感与流畅度。

在生成机制方面，奶龙只会出现在特定的地面区域上，如map0、map1和map2类型的地砖，确保生成位置的合理性与美观性。此外，为避免开局即陷入危险，它不会在距离玩家出生点4格以内的区域生成，同时同一位置也不会重复生成多只奶龙。这一生成逻辑保证了游戏节奏的平衡，也使得地图探索更加紧张刺激。

总体而言，奶龙的随机行动与可爱外观形成鲜明对比，既增加了游戏的趣味性，又对玩家提出了躲避与战斗的挑战，使得整个迷宫探索过程更具活力与变数。

### 幽灵

其原始素材如下：

幽灵是游戏中具有强烈压迫感的敌对单位，其独特的设计与行为使其成为玩家在探索迷宫时最需警惕的存在。外观上，幽灵通常以半透明的漂浮形态出现，使用“ghost”图像表现其幽灵般的状态。当被击败时，它会切换为“blowup”图像，呈现爆炸效果作为视觉反馈。与其他敌人不同的是，幽灵在移动过程中不具备方向性动画，无论朝哪个方向移动，其外观始终保持一致，进一步强化了其神秘与冷峻的形象。

在行为模式方面，幽灵展现出极具威胁性的智能追踪特性。它始终朝玩家当前位置移动，不会因玩家变换位置而迷失方向。更为可怕的是，幽灵具备穿墙能力，完全无视迷宫中的墙体和障碍物，以直线方式不断逼近目标。这种行为配合其恒定的1.5速度，使其成为一个持续且无法预测路径的追击者，一旦被盯上便难以摆脱。

幽灵的生成机制也充满压迫感，它不会随机出现，而是专门在地图的出口处生成。系统会从预设的出口标记（如g、j、h、l、i、n）中随机选取一处生成幽灵，这种设计形成了“守门人”式的突袭效果。当玩家进入新区域、靠近出口时，极有可能遭遇隐藏其中的幽灵，从而打乱节奏、提高紧张感。

总体而言，幽灵凭借持续追踪、穿墙能力与出口埋伏的特性，成为迷宫探索中的高压威胁元素，玩家必须保持高度警觉，在关键节点时刻快速反应，才能成功避开这类潜伏的致命敌人。

### Boss

其原始素材如下：



Boss是游戏中的最终对手，其设计集视觉冲击力与策略挑战于一体，为玩家带来最具张力的战斗体验。外观上，Boss体型远大于普通敌人，占据更多的地图空间，凸显其压迫感。它具备多种视觉状态，包括正常形态、受伤状态以及进入狂暴后的愤怒状态，每种状态配有独立的动画帧，动态变化丰富，展现出强大的气场与存在感。

在攻击方面，Boss具备精准而强力的火球攻击能力。它能够根据玩家当前的位置实时计算发射角度，利用数学函数（如 atan2）确保火球始终朝向玩家。火球在飞行过程中一旦命中玩家或碰撞墙体，即会触发爆炸效果，并伴随具有冲击力的音效，加强战斗的临场感。这种攻击不仅精准，还具备范围伤害，迫使玩家不断移动规避，提升了战斗的节奏感与压力感。



另外Boss可以放出幽灵来抓捕玩家，但这个幽灵可以被人物的声波攻击消灭，除此以外与普通关卡的幽灵无异。

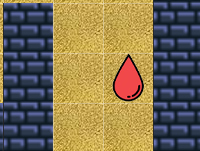
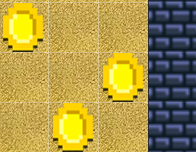
Boss的行为模式同样复杂多变。它并非一味进攻，而是会在攻击的同时评估与玩家之间的距离，适时调整攻击频率与方式。Boss遵循不穿墙规则，但在地图中具备智能移动能力，能够有效寻找进攻位置或拉开距离进行蓄力，形成攻防结合的策略模式。这样的设计让战斗不仅仅是闪避和反击，更像一场步步为营的策略博弈。

此外，Boss关卡是一个独立解锁的挑战区域，玩家需要事先收集特定数量的钥匙才能进入。该区域配备专属的背景音乐和音效，为最终对决营造出庄严而紧张的氛围，标志着整个游戏探索与成长过程的高潮。Boss的登场不仅是技能与策略的考验，更是对玩家操作、判断和心理素质的全面挑战。

## 道具

在游戏中，各类道具为玩家提供了生存保障、能力提升与进阶挑战的关键支持，它们的外观、功能和生成机制均经过精心设计，增强了整体玩法的策略性与探索价值。

血包是基础但重要的回复类道具，其外观通常为医疗包或红心图像，醒目而易于识别。玩家每次受伤会损失生命值，而血包能够直接恢复生命，最高不超过5点的生命上限。一旦玩家角色与血包接触，道具立即生效并消失。每个关卡中血包的生成数量有限，通常只有一个，并被随机安置在可通行区域，为玩家在遭遇强敌或密集敌群后提供关键的生存机会。

金币是游戏中的核心资源，外观为闪亮的金色圆币，有时附带闪光动画以吸引玩家注意。它们分布在地图的各个角落，数量相对较多，可在玩家接触后收集，同时触发“coin”音效。金币不仅用于解锁或购买特殊能力，更是充能“八方向声波大招”的重要资源。系统会实时记录玩家累计收集的金币数，为后续功能解锁提供基础。

蝙蝠声波增强道具则为游戏中的特殊成长类道具，可能以蝙蝠图案或象征性符号呈现。其核心功能是永久提升玩家声波攻击的范围，从初始的1格逐步扩展，最高可提升至4格，使玩家能在更远距离对敌发动攻击。这类道具每关最多生成一个，通常放置在需一定策略或技巧才能抵达的位置，例如狭窄通道、机关附近或需要迷宫切换才能抵达的隐蔽角落，鼓励玩家主动探索和规划路径。

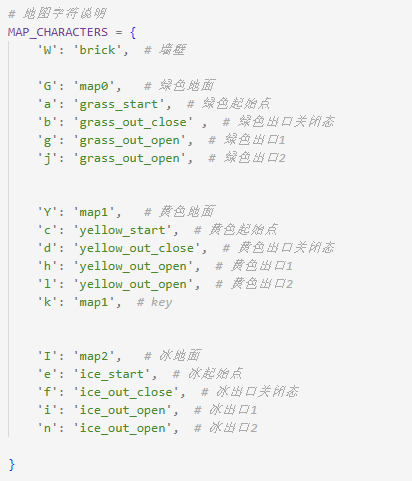
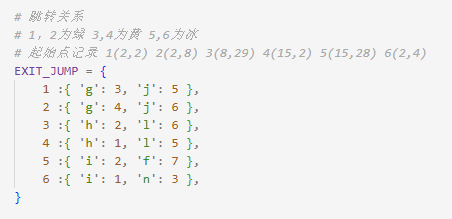
钥匙则是推进流程的关键道具，其外观为传统造型的金属钥匙，通常出现在黄色地面之上，并在地图中以“k”标识。它用于解锁特定的出口或隐藏区域，尤其是在通往Boss关卡时，往往需要事先收集齐钥匙才能开启通路。获取钥匙往往伴随着一定风险或挑战，可能需要玩家绕远路、解谜或战胜特定敌人。

这些道具设计在提供实用功能的同时，也在无形中丰富了游戏节奏与地图探索动机。玩家在追求通关的过程中，不仅要合理规划路线与资源分配，还需在战斗与收益之间做出取舍，从而使整个迷宫探险过程更具深度与挑战性。

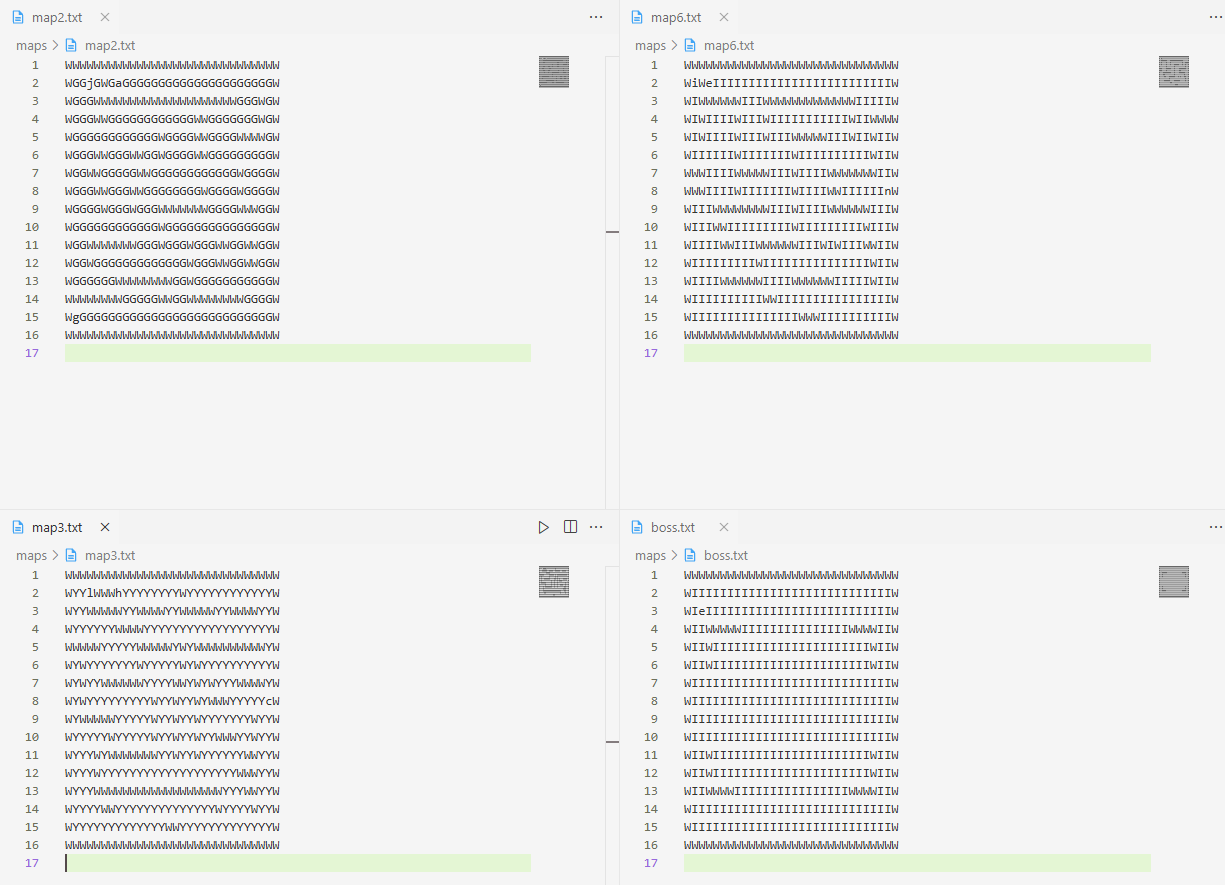
## 地图生成和传送机制

**地图生成和传送机制**是游戏构建世界结构与推动探索的关键系统，决定了玩家在迷宫中的行动方式与关卡推进逻辑。

在config.py中有如下数据结构：

地图字符矩阵：



游戏地图采用**字符文本定义**方式生成，每张地图由文本文件描述，地图中的每一个字符代表一个具体地形或元素（如墙壁、草地、沙地、冰地、传送门等），通过映射关系加载为可视化的砖块贴图。在加载过程中，系统会根据每个字符的位置计算出游戏中对象的精确坐标，并记录特殊元素的位置，如玩家出生点、出口门、钥匙位置等，确保地图布局清晰、功能完整。

传送机制则构建了一个多区域互通的非线性关卡网络。玩家在迷宫中找到并触碰“出口”地砖时，会触发传送逻辑，根据当前关卡编号和出口类型（如g、j、h、l等）跳转到另一个预设关卡，并自动将玩家重定位到新地图的起始点。传送时会播放“level”音效，为玩家提供及时反馈。

每个区域对应两个主关卡，并拥有特定的起点与出口结构：绿色草地区域（关卡1和2）、黄色沙漠区域（关卡3和4）、蓝色冰地区域（关卡5和6）。某些传送门最初是锁定状态，玩家需先收集钥匙才能开启；Boss关卡更是只能通过关卡5的特定出口进入。这种设计打破了传统的线性推进方式，鼓励玩家反复穿梭与探索，寻找最优路线与隐藏要素。

整体而言，地图生成和传送机制不仅为游戏构建了清晰而多样的空间结构，也极大地丰富了玩家的探索路径和游戏策略，是支撑游戏世界构建与挑战递进的重要基础。

# 代码结构与核心逻辑

## 文件结构

游戏采用**模块化设计**，将各类功能合理划分至多个文件中，构建出结构清晰、逻辑分明的项目架构，既有助于代码管理，也提升了开发效率与可维护性。

1. **核心引擎模块**

* main.py：作为游戏主入口，负责调用各模块以完成主循环控制、画面渲染、事件更新与游戏状态管理，整合整体游戏流程。
* config.py：集中管理游戏所需的全局配置与常量，包括屏幕尺寸、游戏状态枚举、地图字符到贴图的映射规则等，为项目提供统一的设置支持。
* map\_loader.py：专门负责地图的读取与加载工作。从maps目录中的文本地图文件中读取字符内容，将其解析为游戏对象并完成地图渲染。
* sound\_manager.py：统一管理背景音乐与各类音效，如拾取金币、传送、攻击、爆炸等，通过封装音频控制接口简化播放逻辑。

1. **角色与行为模块**

* player.py：实现玩家角色的移动控制、攻击操作、碰撞检测、动画切换等核心交互逻辑，是玩家与世界互动的主控单元。
* milk\_dragon.py：定义奶龙敌人的AI行为，包括随机游走、方向判断、动画控制与死亡爆炸效果。
* ghost.py：实现幽灵敌人始终追踪玩家的算法，具有穿墙特性与固定速度移动。
* boss.py：封装Boss敌人的复杂行为逻辑，包括火球攻击、动画状态切换、攻击方向计算等，代表游戏中的终极挑战。

1. **界面与交互模块**

* start\_screen.py：负责渲染游戏开始界面与介绍界面，实现“开始游戏”、“退出”等按钮的可交互逻辑，是玩家进入游戏前的导航界面。

1. **辅助与资源模块**

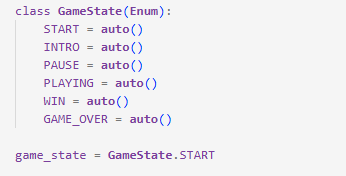
* **Pygame Zero 库**：游戏基于 Pygame Zero 框架进行开发，该框架提供简洁的游戏开发接口，适合快速构建2D游戏。
* **/images,/sounds,/fonts**：集中存放所有图像贴图、背景音乐与音效文件和字体信息，便于资源统一管理与调用。
* **/maps**：保存全部关卡的地图定义文本文件，配合map\_loader.py实现地图动态加载与切换。

## 代码逻辑设计

采用分层架构设计，通过模块化思想构建复杂游戏系统。以下是详细的设计逻辑：

### 状态管理系统

游戏使用枚举类管理全局状态，实现状态机模式：

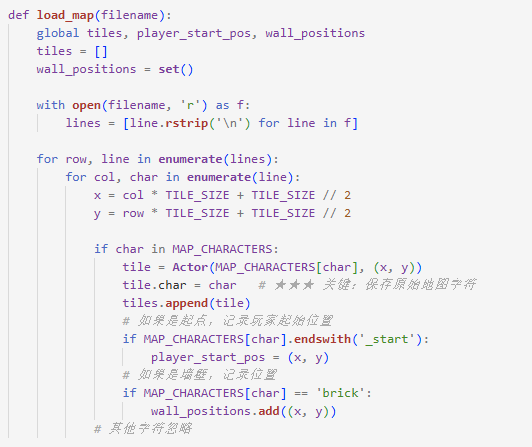


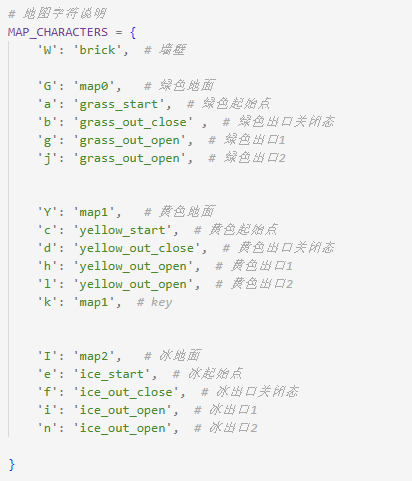
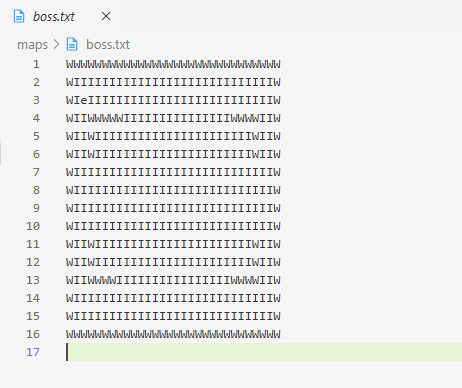
核心游戏循环根据当前状态分发不同逻辑：



### 地图与关卡系统

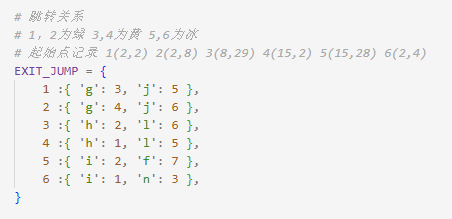
1. **地图加载流程**
2. 从文本文件读取地图布局
3. 将字符映射到游戏对象
4. 记录特殊位置（墙壁、出生点、出口）
5. 构建关卡连接关系





1. **关卡传送系统**

* 使用字典映射出口与目标关卡
* 保存每个关卡的起始点坐标
* 关卡切换时重置相关状态



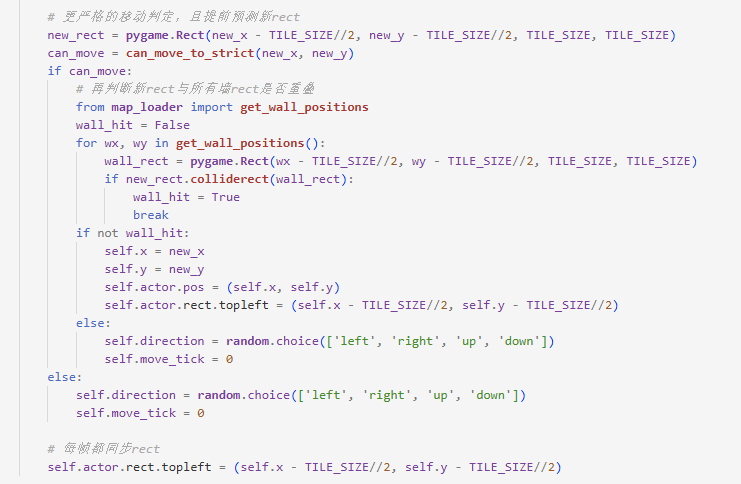


### 角色控制系统

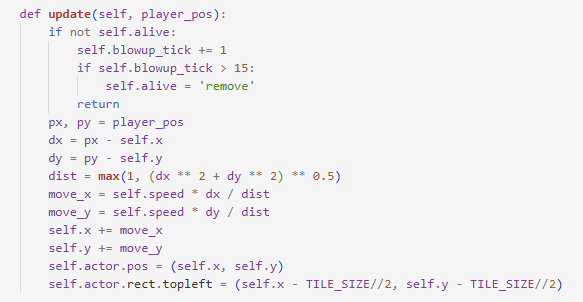
1. **玩家控制**

采用组合式设计，将功能分解为多个专用函数：  


1. **敌人AI系统**
2. 奶龙AI（随机游走）

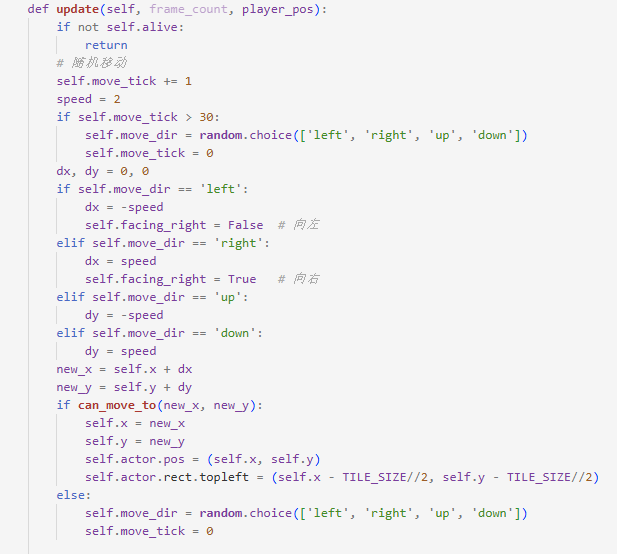


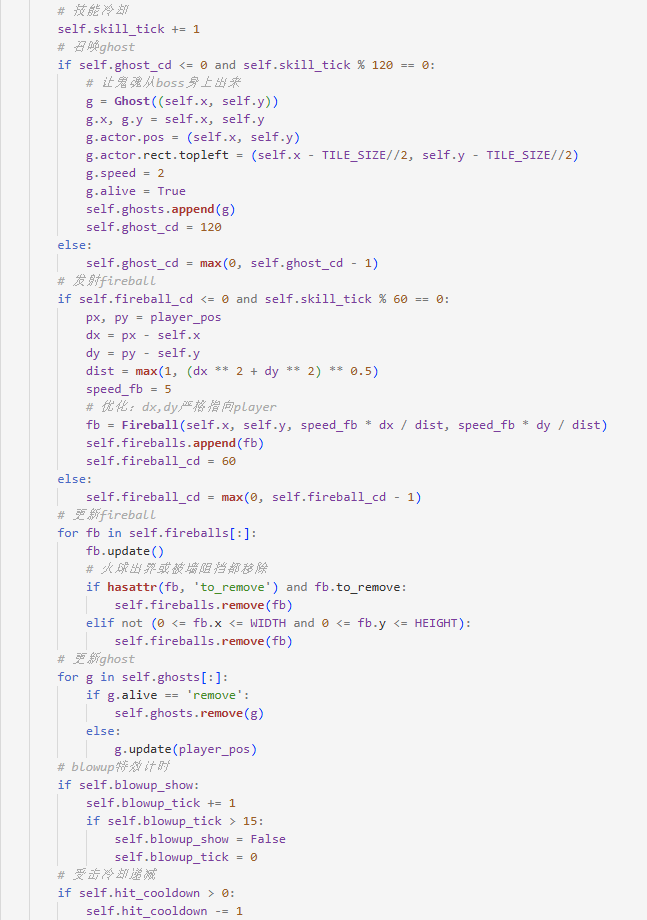
1. 幽灵AI （直接追踪）



1. Boss AI（复杂状态）

* 维护攻击冷却时间
* 根据玩家位置调整攻击
* 控制火球发射角度和速度

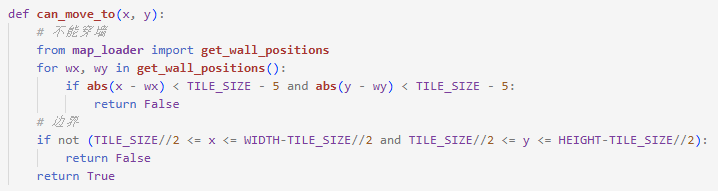




### 碰撞检测系统

多层次碰撞检测：

1. **墙体碰撞：阻止角色穿墙**

****

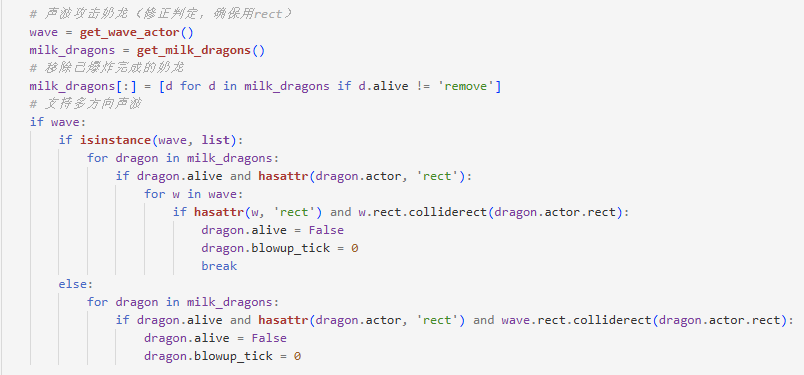
1. **敌人碰撞：检测玩家与敌人间碰撞**

****

1. **道具收集：检测玩家与道具间的交互**

****

1. **攻击判定：声波攻击与敌人的碰撞检测**

****

### 事件系统

1. **输入事件**

* 键盘事件：移动控制和技能释放



* 鼠标事件：菜单交互和界面操作



1. **游戏事件**

* 定时器事件：敌人生成、攻击冷却
* 碰撞事件：damage、恢复、收集
* 状态转换：关卡切换、菜单切换





### 音效与资源管理

**音效系统**



### 模块间通信机制

1. **导出函数**

每个模块提供公共接口供其他模块调用。

1. **数据流向**

* 主模块(main.py)协调各子系统
* 玩家位置信息流向敌人AI用于追踪
* 地图信息流向碰撞系统和渲染系统
* 游戏状态影响所有子系统的行为

这种全面而模块化的设计使游戏代码结构清晰，便于维护和扩展，同时确保游戏运行性能良好。各个子系统独立运作但又密切协作，形成了一个完整、稳定的游戏架构。

# 心得体会

通过开发这个迷宫游戏，我不仅提升了Python编程能力，更深入理解了游戏开发的流程与设计思维。项目采用模块化结构，让我在功能拆分、状态管理、AI设计等方面积累了实战经验，尤其在多敌人行为和传送机制设计中，体会到“看似简单功能，往往隐藏复杂逻辑”的道理。

在开发中，我逐步优化了性能、强化了系统稳定性，也意识到前期规划与清晰结构对项目推进的重要性。这次实践让我真正体会到游戏开发是技术与创意的结合，也激发了我对这一方向更深的兴趣。