**课程设计报告**

——第17小组

1. **团队成员介绍及分工**
2. 成员介绍

|  |  |  |  |
| --- | --- | --- | --- |
| 姓名 | 职责 | 学号 | 班级 |
| **马俊飞** | **组长** | 22090001051 | 22级软件工程 |
| 张巧玲 | 组员 | 22020007141 | 22级软件工程 |
| 朱子欣 | 组员 | 22020007157 | 22级数据科学与大数据技术 |
| 刘畅 | 组员 | 22020007057 | 22级保密技术 |
| 彭若洋 | 组员 |  |  |

2.任务分工

**马俊飞**：（1）（组长）负责项目管理和协调工作，确保开发进度按计划进行。

（2）负责设计和开发玩家角色模块，包括角色属性（如力量、健康），角色升级系统、经验点数等。

（3）负责设计和开发物品模块，包括各种装备、道具、商店物品等 。

（4）负责设计和开发进度存档与读取模块，用于保存玩家的游戏进度。

（5）参与游戏的测试和调试，以确保游戏的各个模块正常运行。

刘畅：（1）负责设计和开发地图与房间模块，包括游戏世界的地图结构、房间定义、区域划分。

（2）负责游戏的用户界面设计。

（3）参与游戏的测试和调试，以确保游戏的各个模块正常运行。

彭若洋：（1）负责设计和开发敌人角色模块，包括敌人属性、战斗系统等。

（2）参与游戏的测试和调试，以确保游戏的各个模块正常运行。

朱子欣：（1）负责解析玩家输入的命令，并执行相应的游戏逻辑。

（2）设计故事背景

（3）参与游戏的测试和调试，以确保游戏的各个模块正常运行。

张巧玲：（1）负责设计和开发命令解析和执行模块

（2）设计故事背景，游戏内容及其各类属性

（3）负责课堂展示PPT

（4）参与游戏的测试和调试，以确保游戏的各个模块正常运行。

**二．课程设计介绍**

1.游戏介绍

游戏名称：魔林探秘

游戏主题：（1）**奇幻主题**：游戏在一个神秘而古老的森林中包括魔法、宝藏和奇幻生物。

（2）**冒险主题**：游戏的目标是探索森林、解开秘密、完成任务

（3）**魔法主题**：游戏中提到幽林是魔法的源头，玩家要寻找 "魔法之心"

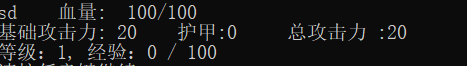
（4）**任务和解谜**：玩家必须完成各种任务，解锁房间的谜题，这增加了游戏的复杂性。

游戏背景：在一个遥远的魔法王国，有一个神秘而古老的森林，被称为”幽林“。幽林被认为是魔法的源头，充满了千奇百怪的生物和宝藏。然而，进入幽林的道路却不是轻松的事情。**这片森林被一系列神秘而危险的房间所保护，每个房间都有其独特的挑战和宝藏。**现在，你是一位冒险家，你踏上了前往幽林的旅程，但必须在克服一系列挑战和与各种奇幻生物互动的同时，解开这个古老森林的秘密。你的目标是找到幽林深处的传说中的“魔法之心”。然而，**你必须面对各种各样的任务、危险，以及与NPC角色的互动，才能成功获得这颗宝石。**你的命运和整个王国的命运都取决于你的勇气和智慧。

2.功能介绍

**（1）.查看角色状态**

\*查看角色各类属性（如：等级，经验条，攻击力等）



**（2）.查看角色背包及装备**

\*展示背包内物品，及穿戴的装备及其属性

\*装备替换和使用



**（3）.房间探索功能**

\*检查并拾取房间内可得物品

\*探寻房间内敌人

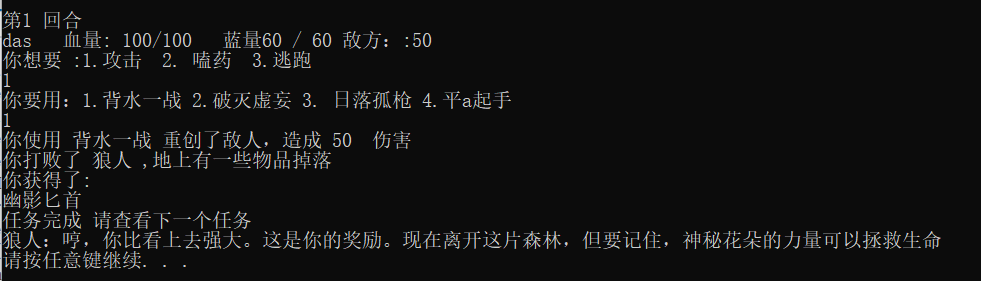


**（4）.战斗相关功能**

\*选择战斗对象

\*战斗时技能选择及药品使用

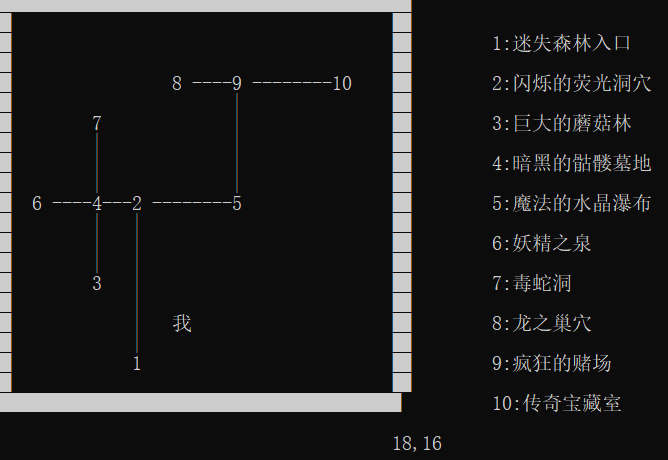
\*逃跑功能



**（5）.地图功能**

\*展示地图

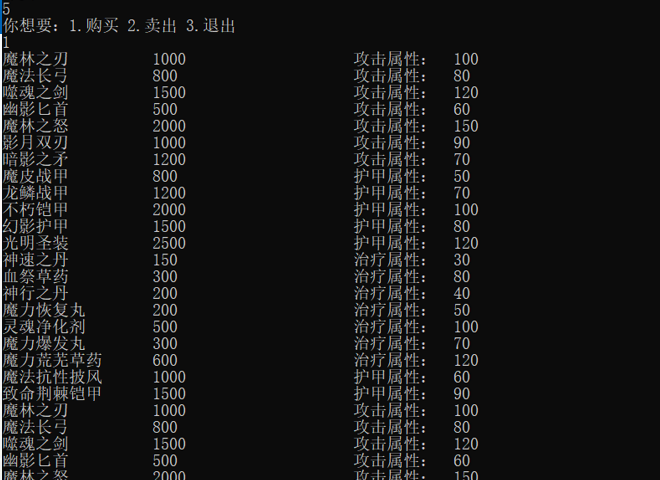
\*利用控制台操作，根据当前位置在操作地图上进行自由移动



**（6）.商店功能**

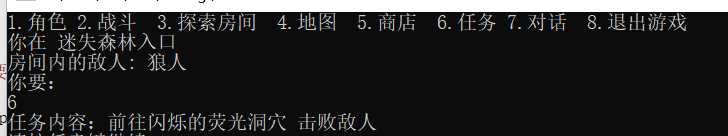
\*展示各商品及其属性

\*可根据数量和商品名称购买商品

\*可根据背包内物品进行售卖

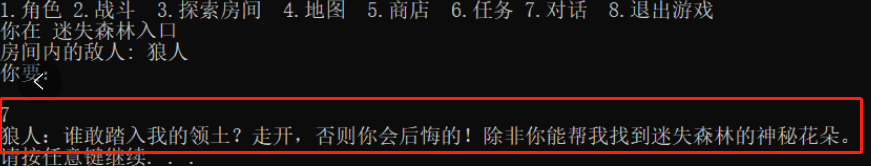
**（7）.任务功能**

\*查看任务主线并接受任务



**（8）.NPC交互功能**

\*通过对话形式提供线索，完成相应任务



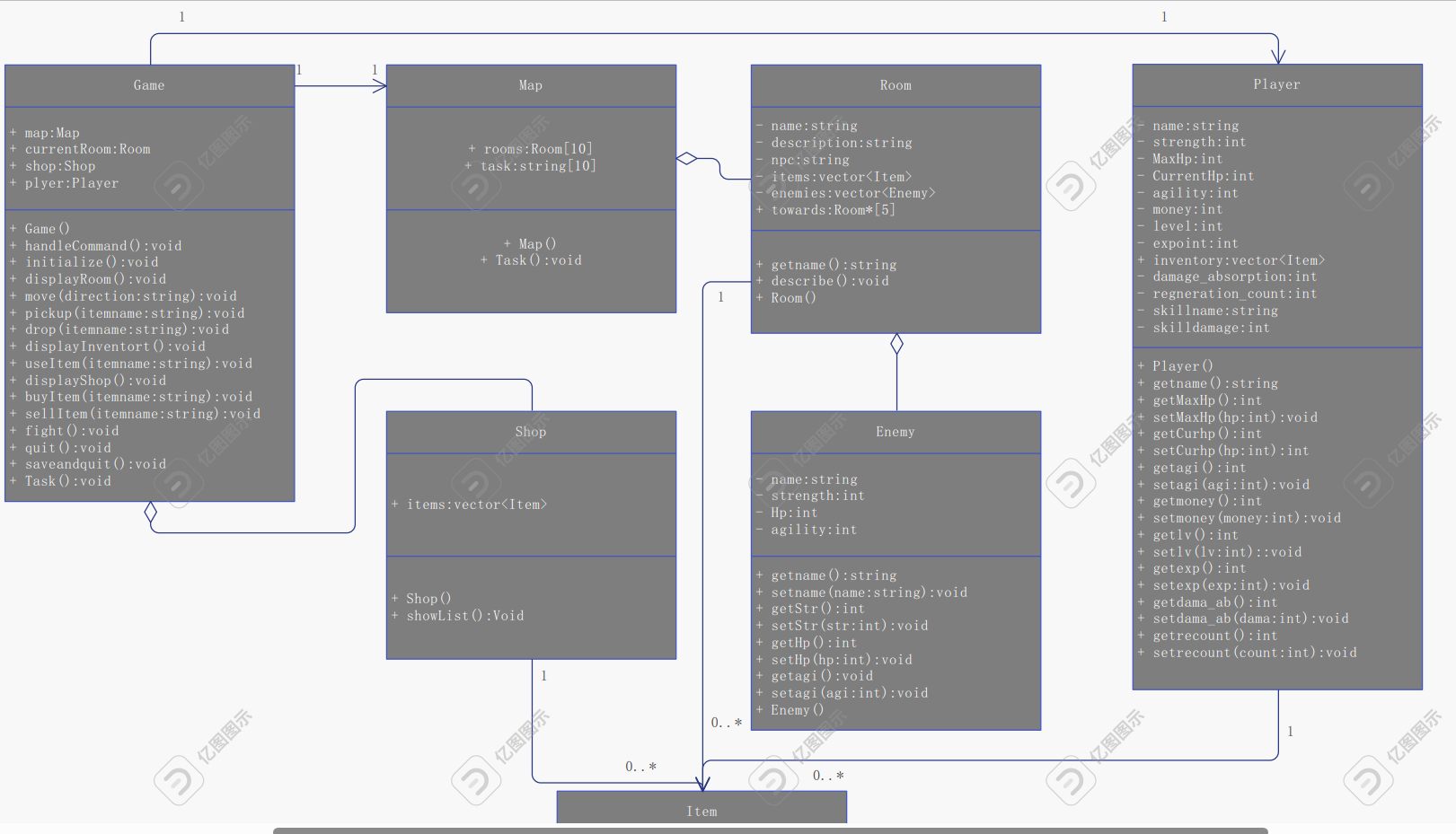
**（9）.存档及读档功能**

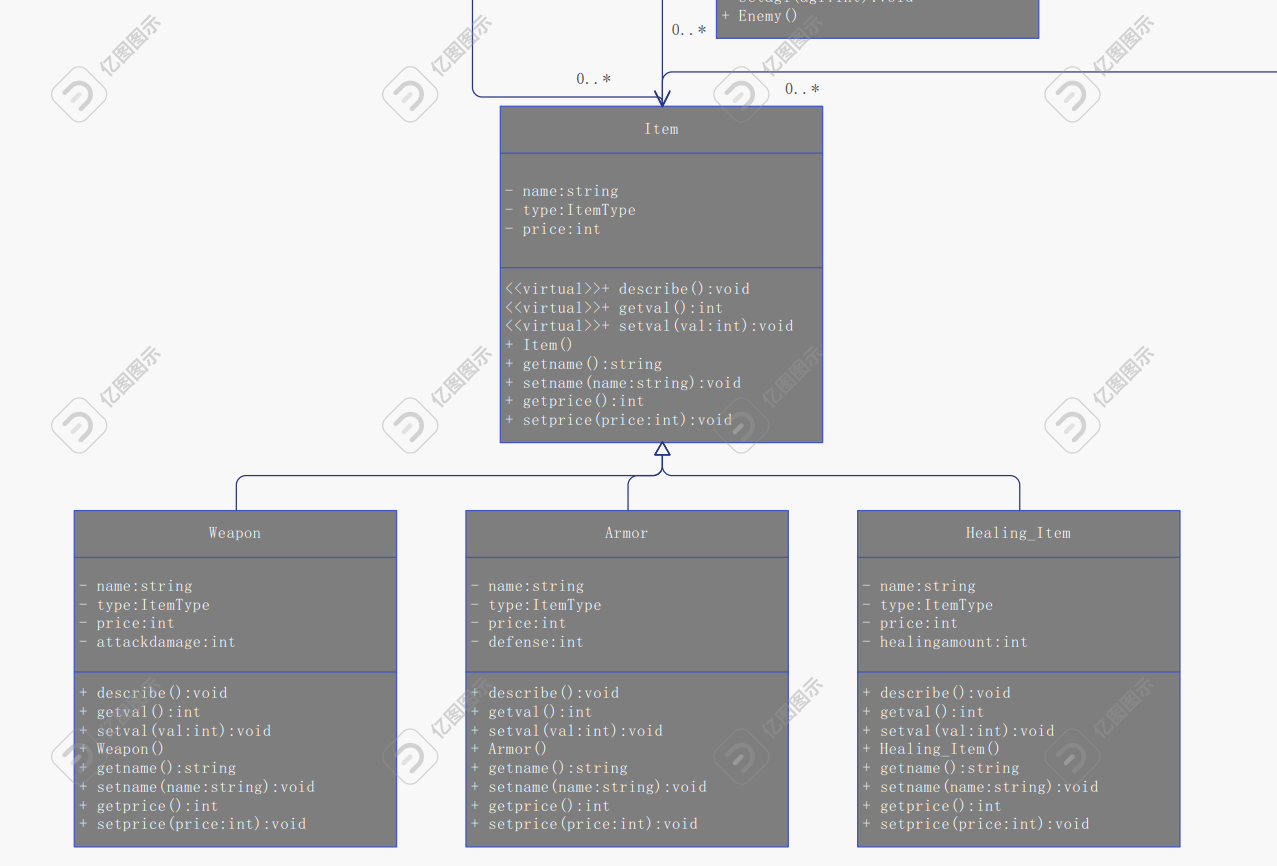
\*保存当前游戏状态，任务进度及装备

\*读取存档，恢复游戏状态，继续进行游戏

1. **UML类图介绍及分析**

UML类图如下：





**UML图介绍及分析**：

首先设置Item类 Enemy类 room类 Player类 Shop类中各种属性和简单方法，room类通过聚合形成Map类 Item类设置三个子类，Player Room Shop都会使用到Item类，并在Room类中设置分别装有物品与敌人的容器vector，再Player Shop中设置可装有Item的容器，在player中使用了map容器便于储存一些具有对应关系的值，如技能名字与伤害，并通过迭代器进行遍历访问，设置一些处理对象关系的稍复杂函数 ，在Item基类中设置纯虚函数，使Item作为抽象类，泛化的三个子类继承父类的一些属性，并分别拥有各自独特的属性，父类中的纯虚函数在三个子类中分别实现不同的功能内容，最后可通过父类的指针直接调用在特定子类中实现的功能，提高了代码的重用性和简洁 最后在Game类中对各类实现同意调用进行各种功能的实现。

**四．设计模式**

单例模式 ：Game类只有一个实例

**五．项目中遇到的问题和解决方案**

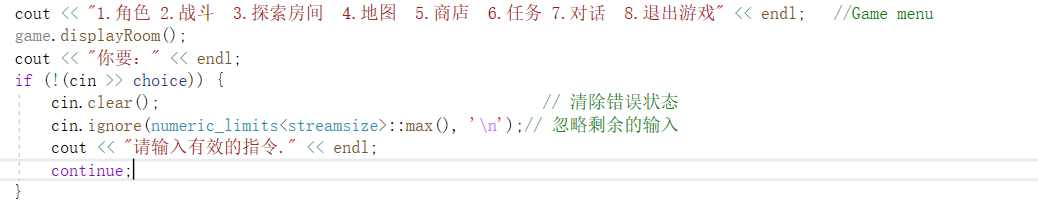
1.在游戏数据的保存和读档中，要使所需的重要数据进行保存，需将各种变量通过文件操作输入到save.dat文件中，而要保存的数据不仅包含string int bool 等基本类型的数据 也包括自定义的数据类型 如枚举类型ItemType，物品类Item。对于前者，我们可以通过类型转换 static\_cast 先使其转化成int类型进行保存，读取后再转化成ItemType类型的即可。对于后者，虽然我们可以将所需保存的数据通过get方法一一输入到文件中，但不免麻烦与冗杂，一个解决办法是，在Item类中，对<<与>>两个输出输入运算符进行重载，作为Item类的友元函数，在函数中编写好我们对我们所需的特定数据进行关于文件的读取操作，这样一来，可减少代码冗杂，简便实现。同样的，这个方法也可适用于其他类的数据保存中

2．在程序中，我们使用到了vector与map等容器与相关操作。对于vector来说，有时可能不便于使用类似数组方式的下标或索引来访问某个元素或遍历整个容器，还可能出现越界的错误。我们可以通过其他方式进行遍历和访问操作。第一个是使用for-each循环，即范围for循环。特点是，这种方式无需使用索引或迭代器来迭代整个容器，且语法简洁，易于理解和书写，并避免了越界访问的错误，无需手动控制迭代范围 例如，遍历访问player对象中的Item类向量，可使用for(Item& item: player.inventory)。第二种方法便是使用迭代器来遍历 如

for(auto it = player.inventory.begin();it!=player.inventory.end();++it)，需调用容器的begin()与end()成员函数来指定迭代的起始位置和终止位置，好处是，迭代器it会逐个指向库存中的物品，便于对其进行操作或访问其属性。如，当我迭代到某个对象时，发现正好满足某些条件，如扔掉它，便可以直接使用erase(it)来删除迭代器所指的元素 it=erase(it) 便可指向删除元素的下一个有效元素继续遍历 十分方便，需注意的是，在调用erase之后，被删元素之后的所有元素位置都会向前移动，并且容器大小也会改变

3.在程序某些功能实现的过程中，发现一些奇怪的bug，比如，在做完任务之后理应的任务没有自动刷新，在使用替换装备之后发现原来的装备还在使用中，这一类问题以装备问题为例。在player中，使用的武器和装备单独在两个equippedWe equippedAr 中，使用的过程是将要使用的item赋给对应的equipped对象，并设置item和对应的equipped对象的是否使用bool数，装备替换是再将要替换掉的装备的bool数设置回来。一开始用的equippedWe equippedAr只是简单的两个类的对象，如此一来，在装备替换中的bool的设置，值传递无法传回一开始使用的item，导致item没有卸下，这一类问题可使用类指针轻松解决

4.在程序设计时，许多动作功能，例如查看背包，使用物品等都是通过输入一些选项来触发的，使用最多的便是数字选项。但是，如果在输入选项时输入一个字符串，便会引发错误，使cin处于错误状态，使程序陷入崩溃或无限循环。可以做如下处理



cin >> choice 尝试从标准输入流中读取一个值，并将其存储在 choice 变量中。if (!(cin >> choice))：这一行检查 cin >> choice 是否成功。如果成功，cin 的状态位将保持正常，表示输入是有效的，然后代码会继续执行。如果输入无效或不符合预期（例如，用户输入了非数字字符），cin 的状态位将被设置为错误状态。cin.clear();：如果 cin 处于错误状态，此行代码用于清除错误状态，以便可以继续从标准输入中读取后续的输入。cin.ignore(numeric\_limits<streamsize>::max(), &apos;\n&apos;);：如果发生了输入错误，这一行代码用于忽略掉输入缓冲区中的剩余内容，直到遇到换行符 \n，以确保之前的错误输入不会影响后续的输入操作如果发生了输入错误，continue 语句将控制流返回到循环的开始，以便用户可以重新输入有效的内容。如此便可确保接受有效的输入，并在输入无效时进行适当的错误处理，以避免程序崩溃或陷入无限循环

**六．总结与展望**

**1.总结：**

（1）团队成员介绍及分工： 我们的团队由五名成员组成，每位成员都有明确的分工和责任。组长负责项目管理和协调，确保项目进展顺利。其他成员分别负责不同的模块。

（2）课程设计介绍： 我们的课程设计是一个名为"魔林探秘"的奇幻冒险游戏，以神秘的幽林为背景。游戏主题包括奇幻、冒险、魔法和解谜，玩家需要探索幽林、解锁房间的谜题、完成任务，并寻找"魔法之心"。

（3）功能介绍： 游戏具有多种功能，包括查看角色状态、背包管理、房间探索、战斗、地图导航、商店购物、任务完成和NPC交互。此外，还支持存档和读档功能。

（5）UML类图介绍及分析： 我们的团队使用了UML类图来设计和组织游戏中的类和对象。你详细介绍了各个类的属性和方法，并说明了它们之间的关系，例如继承、聚合等。

（6）设计模式： 我们在报告中提到了单例模式在Game类中的使用，这有助于确保只有一个游戏实例存在。

（7）项目中遇到的问题和解决方案： 我们在以上内容中列举了在项目开发过程中遇到的一些问题，如数据保存与读档、容器操作、装备替换和输入处理等，并提供了解决方案，例如类型转换、迭代器使用、类指针等。

**2.展望：**

（1）进一步的测试和调试： 在完成项目的基础上，我们将继续进行测试和调试，确保游戏的各个模块正常运行，发现并修复潜在的错误和问题。

（2）优化用户界面： 进一步优化用户界面，增强游戏的可玩性和用户体验。

（3）扩展游戏内容： 将会考虑是否可以添加更多的任务、房间、物品和故事情节，以增加游戏的深度和复杂性。

（4）支持更多平台：我们会考虑将游戏移植到不同的平台，如移动设备或Web浏览器，以扩大受众群体。

（5）反馈和改进： 计划增加反馈的实现，能够接受用户反馈，了解他们的需求和建议，以不断改进游戏并发布更新版本。

（6）继续学习和改进： 我们团队打算继续学习新的开发技术和设计模式，以不断提高自己的能力，并在未来的项目中应用新的知识和经验。