编号：

|  |  |
| --- | --- |
| 设计报告文档成绩 |  |
| 设计作品分数成绩 |  |
| 课程设计成绩 |  |

**程序设计课程设计文档**

**题 目 ： 和平岛（游戏）**

**姓 名 ：**

**学 号 ：**

**专业班级 ： 计算机一班**

**2023年8月26日**

**一、题目意义和设计思想**

|  |
| --- |
| **1、题目意义**  ■ **警示教育**  《和平岛》（游戏）以两个海岛之间相互杀戮和冲突，双方两败俱伤的场景，向人们宣示：对抗中没有胜利者，战争只会给人类带来灾难，良知、鲜血和生命呼唤人类和平。游戏蕴含现实意义，契合学院老师对设计作品的目标要求。  ■ **休闲娱乐，丰富人们精神生活**  ■ **提升自身业务素养**  **⑴** 检验巩固自己《高级语言程序设计》的学习成效，查漏补缺，对所学知识拓展、深化和系统化。  **⑵** 培养和增强计算思维，锻炼自己在不同领域运用编程解决实际问题的能力。  **⑶** 掌握C程序开发的全过程，熟悉C语言的语法规则，面向过程的编程思想。  **⑷** 养成严谨的编程习惯和简洁优雅的程序风格。  **⑸** 掌握文献查阅方法，提高学习能力。  **2、设计思想**  **面向过程思想：**  函数概况：主要通过gameTwoPeople（游戏运行的统筹总函数）、excute（玩家操作主函数）、attack（攻击主函数）、placeArm（出兵主函数）、placeBridge（架桥、拆桥主函数）及其附属函数实现了游戏的平稳运行。(五者结构关系详见：**三**、实现的主要功能和系统结构)  **各核心函数及其附属函数：**  **⑴** gameTwoPeople:游戏运行的总函数，用于统筹所有游戏执行函数。  ① creatBackground:初始化岛屿、桥梁、玩家。  ②decideChoice:执行掷色子模块，返还胜利玩家编号给gameTwoPeople。  **⑵** excute：对接attack、placeArm、placeBridge三大操作函数、作弊函数和“屯”模块，实现玩家游戏操作。  printIsland:展示游戏当前战况，重置图层。  **⑶** attack:接收胜利玩家编号。统筹攻击者、受袭击者检索、攻击执行以及核打击函数，实现攻击模块。  ① sortAttacker:接收胜利玩家编号。根据中间岛所属权状态， 分情况判断近战、远程兵种是否可用于攻击，将结果通过信息验证机制返还给attack。  ② sortVictim:接收受攻击岛屿编号、嘲讽状态。根据嘲讽状态和球哥状态检索可受袭击者，检索结果通过信息验证机制返还给attack。  ③ attackExcute:接收岛屿编号、位置编号、伤害数值。对相应位置单位执行伤害结算，判定单位是否死亡并执行相应的初始化程序。  ④ nuclearStrike:接收胜利玩家编号。内置岛屿判断和交互选择，对编号岛屿上每个有单位的位置调用attackExcute执行秒杀伤害。  **⑷** placeArm:统筹出兵岛屿检索和岛屿具体位置检索，将兵种数据附属于岛屿。核弹可单独调用nuclearStrike执行核打击。实现出兵模块。   1. sortSoldierIsland:接收胜利玩家编号。进行普通兵种可放置岛屿检索并通过信息验证机制返还给placeArm。   ② sortHouseIsland:接收胜利玩家编号。进行房子可放置岛屿检索并通过信息验证机制返还给placeArm。  ③ sortPosition:接收岛屿编号。挨个检索并返回兵种在选定岛屿的放置位置给placeArm。  ④ houseSoldier：接收胜利玩家编号。调用sortSoldierIsland检索小兵可放置位置，执行自动出兵（可放弃）。  **⑸** placeBridge:统筹架桥、拆桥检索函数，实现基础操作模块。  ① sortBuildBridge:接收胜利玩家编号。检索可架的桥梁并通过信息验证机制返还给placeBridge。  ② sortBreakBride:接收胜利玩家编号。检索可拆的桥梁通过信息验证机制返还给placeBridge。  **⑹** cheatCode:接收胜利玩家编号。输入对应指令，快速放置房子和核弹，加快游戏进度。  **主函数设计：**  **8196df76b342ddbd4d1be6713133667**  65273ef74f81056cb1e6acdfd0a1b8c  主函数设计了两级菜单，首级菜单提供游戏入口，二级菜单中提供了游戏玩法介绍和双人模式入口，两者为以后的项目功能扩充和游戏更多模式扩展提供空间。同时，main.cpp文件中预留peopleAi空函数，用于未来的人机对战开发。 |

**二、采用的主要技术、遇到的难点和解决方法**

|  |
| --- |
| **1、主要技术**  **⑴** C语言面向过程，用于程序流程实现。  **⑵** C文件流与输入输出流：使用FILE指针打开玩法介绍、彩蛋的文本文档，输入输出实现游戏交互。  **⑶** 宏定义：使用宏定义设置岛屿初始兵种单位上限，便于后期调整初始兵种上限，提高了程序的可维护性。  **⑷** 图形化：调用easyx库进行图形化输出，提升玩家游戏体验。  **⑸** 头文件：项目整体超过1500行、二十多个函数。编写help.h头文件总体定义了结构体，声明各.cpp文件之间的主要调用函数和变量，采用#ifndef+#define+#endif避免了重复编译，解决了不同.cpp文件之间的声明重复而琐碎的问题。  **2、遇到的难点和解决方法**  **⑴** 编译时遇到"multiple definition of‘attack(int)’”等错误。  **解决方法：**阅读报错并查阅相关资料后，发现是因为编译时遇到了对同一函数的重定义，出现这种错误往往是因为存在.cpp相互#include。对程序进行检查后，发现是因为我的游戏底层搭建.cpp引用了attack.cpp文件，而C语言编译器会编译所有.cpp文件，这造成了attack.cpp下的函数重复定义。通过查询 C 语言语法相关资料，我了解到头文件的运行逻辑，重新创建了help.h头文件集中处理函数和变量的声明，并使用了条件编译#ifndef+#define+#endif规避重复编译，解决了函数重复定义的问题。  **⑵** 在判断岛屿是否为空岛时，房屋需要被检索到，但房屋本身又不占用岛屿单位空间。  **解决方法：**在房屋存在时，额外进行岛屿单位上限+1，当房屋销毁时上限-1，这样等价于房屋没有占用岛屿单位。实现了房屋占位检索并不会影响其他兵种的正常岛屿单位。  **(3)** 放背景音乐时出现了无报错但无音乐的情况。  **解决方法：**我首先详细排查了相关库的链接和播放器状态，确认不是环境的问题。接着我四处搜寻资料，在一个论坛上面了解到可能是网易云音频的问题。网易云的音频自带封面，干扰了PlaySoundA的数据读取。我重新在QQ音乐上下载歌曲并进行格式转换后成功播放背景音乐。 |

**三、实现的主要功能和系统结构**

|  |
| --- |
| 83fe353ada6d2b919be99430e964661**1、源码架构**  考虑到程序后期的开发和维护，源码采用分文件编写的方式。help.h声明游戏所需的头文件、共同调用的函数和变量，定义岛屿等核心结构体。其余三大操作函数attack、placeArm、placeBridge及其相关函数分别构成对应的.cpp文件。  **2、主要函数调取关系**  main  gameTwoPeople  creatBackground  decideChoice  printlsland  excute  printDice  attack  cheatCode  placeArm  placeBridge  sortHouselsland  nuclearStrike  sortBuildBridge  attackExcute  sortBreakBridge  sortSoldierlslandd  sortVictim  sortPosition  sortAttacker  **3、关键函数与游戏流程对照**  **⑴ gameTwoPeople：**统筹游戏所有函数，实现游戏基本框架：  开始  初始化  是  掷色子  操作执行  否  结束判断  其中crearBackground、decideChoice和excute分别为初始化、掷色子和操作执行的函数实现。  **⑵ excute:**统一调度三大操作函数和“屯”模块，扩充作弊码功能。  **作弊码**  **出兵操作**  **执行操作**  **基础操作**  **攻击操作**  其中attack、placeArm、placeBridge、cheatCode分别为攻击操作、出兵操作、基础操作和作弊码。  **4、图形化**  核心调用easyx库函数。同时，项目自定义了createButton、clear、printIsland等函数对按钮制造、操作页面初始化、岛屿战况输出等图形操作进行了集成封装。总体实现了较为优良的图形页面效果。 |

**四、核心算法描述和相关技术说明**

|  |
| --- |
| **1、兵种设计**  项目最初采用各类兵种对应的结构体，运用数组储存具体兵种单位，通过父类指针与岛屿挂钩（如图1）。  fee16bccef646e5dbc895e7349797d23467a609dd3ea7ebe343e53ba7e4daa 图1 图2  此处的父类指针用结构体+int的方式实现，首先一个struct包含各兵种指针，在程序运行时用veri（int型）判断具体兵种。该方法时间复杂度低，代码可读性高。  但上述方法用数组储存兵种，空间复杂度太高，并有小概率出现数组溢出，程序不稳定。故而采用链式结构储存兵种，然后使用void型指针实现父类指针将岛屿单位与具体士兵挂钩（如图2）。该方法大幅降低了空间复杂度，但节点的创建删除相对提高了时间复杂度。  为了从根本上优化程序流程，我放弃了通常的兵种信息储存。在岛屿单位结构体中增加HP成员变量，岛屿上直接进行伤害结算（如下图）。这样减掉了具体士兵结构体，提高了整套程序的时间效率和空间效率，大幅精简了代码，提高了程序质量，形成了最终版本。  1697206021054  **2、信息反馈**  该项目通过在检索函数中加入数组形参，记录正确信息反馈给上级函数。上级函数处理反馈信息，运用createButton函数产生对应的操作按钮，并通过while循环+peekmessage检测点击实现页面按钮交互。  **3、动态页面**  游戏通过clear和printIsland两个函数实现对玩家的操作反应进行实时响应。clear通过背景图覆盖操作区实现感官上的动态页面，响应速度优良，用于一轮操作中途的动态页面实现；printIsland则对所有图层进行清空再重新绘制，可以避免clear造成的图层堆叠降低性能，但响应速度较慢，用于一轮操作后的总处理。 |

**五、总结和体会**

|  |
| --- |
| **1、技术亮点**  **⑴ 打破思维惯性。**通常兵种类游戏采用面向对象的编程思想，通过类针对各兵种进行处理。然而在综合考量游戏体量、C语言程序特性后，我打破了兵种设计的思维惯性，将兵种直接依附于岛屿位置，精简了代码，同时大幅降低了程序的时间复杂度和空间复杂度。  **⑵ 重视编程的规范性。**本项目代码书写规范，我遵从驼峰命名法的原则命名，结构体名每个单词首字母均大写，函数和变量名第一个单词全部小写，其他单词首字母大写；在代码架构上，我将所有结构体存放在了help.h 文件中，并在相应的cpp文件中实现三大操作模块的相关函数定义。编程的规范性使得程序的健壮性大大提升，为程序日后的拓展和维护带来了很大的便利。  **⑶ 图形化编程。**我们以往的编程通常局限于字符模式。为了增强游戏体验，“和平岛”引入图形化编程，大幅提升了玩家的视觉感受。同时，游戏通过详细的交互信息、Q版的页面和人物设计给人以良好的游戏体验。  **2、选题依据和对作品的自我评价**  **⑴ 以踮起脚尖摘桃子的高度确定目标。**我确定的目标是：在保证质量的前提下，作品有1000余行语句的规模（实际有1500多行）；程序比较复杂，有一定难度；技术上有创新。为达到这个目标，我和多数同学一样，选择了游戏程序设计。理由是，游戏需要的程序语句多，比一般管理软件复杂；有创新空间，能满足学院老师对这次设计在作品规模、难度和创造性上的要求。同时，编程的本质就是模拟游戏，游戏程序设计方法和技术可移植到各种类型软件。  **⑵ 以“工匠精神”雕刻《和平岛》。**《和平岛》整体上精心构思，每一步精雕细刻。遇到难点，读教材，查文献，悟原理，上机测试找原因，凭自己独立思考找到解决方案，实现技术创新。全程时间，因为和历时30多天的学校机器人实验室入室考核（我在考核之列）重叠而无法计算。课程设计各阶段时间占比，纸上设计约占课程设计总时间的20％，查资料思考问题占30％，上机输入调试占50％。  **⑶《和平岛》是我的处女作和满意作品。**《和平岛》架构设计合理，程序编写规范标准，简洁流畅。经反复测试，其稳定性、可延展性等各项功能、性能均有很好的表现，是一件设计有难度，程序规模较大，功能、性能优越的满意作品。  **3、体会**  程序设计是脑力、耐力和创造力的合成艺术，成果是“艺术品”。 程序设计时，要精心构思，认真探索，积极改进创新。本次程序设计中，我摆脱了兵种设计的思维惯性，将兵种直接依附于岛屿位置，大幅提高了程序的时间效率和空间效率。同时精简了代码，形成整套程序简洁、高效的风格，确保了软件整体质量。  分层实施时，要标准化编程，增强代码可读性，提高编程效率，减少不必要的时间浪费。要有铁杵磨成针的耐心，反复琢磨，逐步优化。要有细致入微的洞察力，快速锁定bug的逻辑源码。同时提前设想极端情况，提高测试效率。 |