项目开发报告

**课 程 名 称**: C++综合实验

**课 程 代 码**: 190901029

**项 目 名 称**: 坦克动荡游戏

**年 级/专 业**: 2019级/计算机科学与技术

**学 生 姓 名**: 刘唐

**学 生 学 号**: 3120190971181

评 定 成 绩

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 平时表现  （20） | 开发能力  （30） | 项目工作量  （20） | 报告质量  （30） | 总 分  （100） |
|  |  |  |  |  |

教师签名： 20 年 月 日

坦克动荡游戏

**摘 要**

坦克动荡游戏是一款简约有趣的坦克对战游戏，游戏场景设定在一个随机生成的小迷宫中，游戏双方控制自己方坦克攻击对方，直至一方坦克爆炸为止，在游戏中坦克可以连续发射多颗子弹，需要小心的是子弹打到墙上会反弹，反弹的子弹可能打爆自己的坦克，所以千万要选好角度再发射子弹，不然会反射到自身。

此款游戏包含动态游戏菜单、人机大战、双人大战、自动寻路、寻找最短路径和子弹反，等功能。

【**关键词**】Gdiplus；碰撞检测；自动寻路实现

**1绪论**

“坦克大战”，游戏进行时，敌方坦克攻击，玩家控制自己的坦克保卫家回，坦克可以发射子弹互相对打，但是传统的坦克大战游戏，坦克的移动规则限于上下左右四个方向，而不能任意方向自由移动；同时子弹也只能横织四个方向进行攻击。为增加游戏乐趣，本次实验使用了Windows下的C++语言，配合Windows API开发一个坦克动荡游戏。

**2 总体方案设计**

2.1 设计任务及要求

本次设计的主要功能要求：

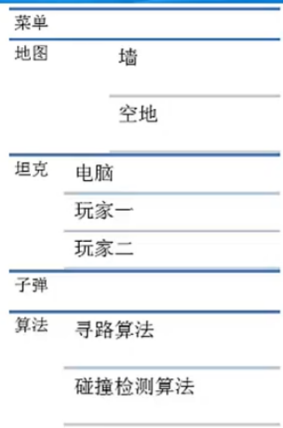
1. 菜单：通过Gdiplus加载绘制位图图片背景图片，菜单选项还需添加鼠标相应。

图 2.1.0 主要功能要求

1. 可移动对象设计：提供移动旋转等相关函数
2. 地图设计：地图单元思想、单元又分为墙和空地、碰撞检测、路径搜索等。
3. 坦克设计：玩家一、玩家二、电脑控制坦克、坦克不能穿墙。
4. 子弹设计：移动方向、碰撞墙壁后反弹、一定时间后消失。

④ 算法设计：寻路算法、碰撞检测算法。

**3 详细设计**

我将游戏框架与各个对象功能的原理和具体实现方法一一呈现出来，并给出部分流程图和关键代码。

3.1 场景渲染详细设计

3.1.1 游戏框架

游戏框架好比建筑的承重柱与衡量，其余各种设计都在这框架中运行。一个好的程序框架不仅会使程序的设计显得更为清晰简洁，还会增加优化空间，提高“装修”效率。

此次游戏窗口不需要菜单与工具栏。清除ChildView相关代码，自己创建一个窗口。

后续的开发会增加很多代码来实现游戏的各种功能。如果都写在同一个文件中，会显得非常拥挤，所以这里设计一个CGame类，把游戏相关功能拆分开来。本程序还将设计很多类，每个实现不同的功能并通过CGame类“串联”起这些类以实现完整的游戏程序。这样的好处是结构清晰，易于扩展，方便理解。

CGame类代表“整个游戏”对象，所有游戏中的其它功能模块皆有本类进行管理。

图 3.1.1.0 CGame类

其他功能又分为几个大类，如图3.1.1.0。

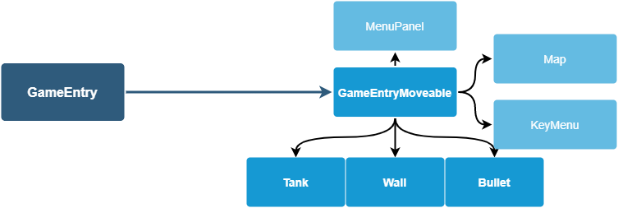
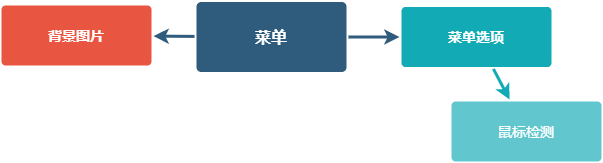
游戏运行时总是一帧一帧运行的，由此则需要一个定时器消息来实现。在CGame类的公有方法中有一个重要的成员函数：EnterFrame 就是处理每一帧的函数，它将会作为每一帧的处理函数，调用绘图函数与逻辑处理函数。

图3.1.1.1 游戏对象的继承关系

游戏中的具体对象的父类名为CGameEntry，由它继承一个子类（可移动对象类），最后就由它继承出各个具体的游戏对象。具体的游戏对象有六个，具体继承关系如图3.1.1.1

3.1.2 菜单

菜单是游戏的门面，其分为显示背景图片与显示菜单选项。在显示菜单选项中，实现鼠标移动到选项上后，显示特殊效果

（人机大战显示机器人，双人大战放大图片以表现突出效果。）

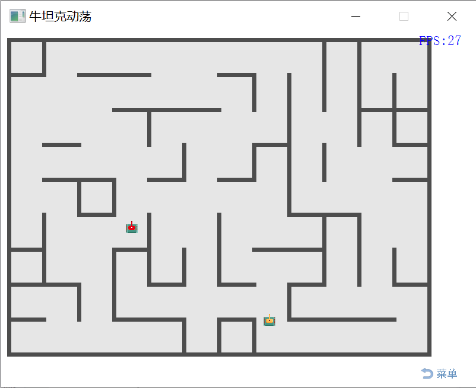
图3.1.2.0 菜单功能

显示背景图片即调用CGameMenuBackground中的Draw函数，不多言。

菜单选项除了画出按钮对应的图片外，为了实现鼠标移动到这个图片上后转换成其它图片或绘制另一属性图片，以及点击后的处理，对其添加了鼠标相应。添加鼠标响应函数中，有一个鼠标移动的消息，在此消息中，直接把鼠标消息传给CGame对象，在CGame对象对应函数中又调用Panel对应的函数实现相应具体功能。

3.1.3 地图、坦克

生成地图的逻辑是单个方块随机生成周边的围墙，中间是空地。多个方块组合铺满地图范围，地图边界都是围墙。地图的绘制时间为游戏启动，选择开始游戏后。

坦克分为人类操控坦克与机器操控坦克，机器操控坦克需要实现自动寻路与射击算法。坦克的绘制时间与地图的绘制时间一致。所以在调用GameRunDraw时，即游戏启动，选择开始游戏后。生成效果如图3.1.3 。

3.1.3.1坦克的控制：

中心思想是：处理按键消息。

在第二个逻辑处理函数中创建处理按键的代码：

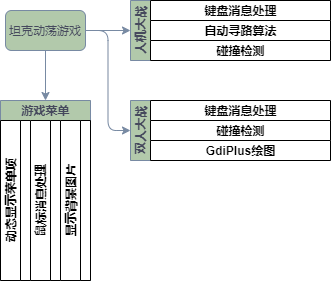
if (KEYDOWN(VK\_LEFT)) {// 左方向键按下

m\_player01.RotateLeft();// 玩家向左旋转}

3.1.4 自动寻路

通过广度优先算法，查找最小路径，具体算法不赘述了。

图3.1.3 地图与坦克的生成效果

**4 结论**

本次实验初步了解了消息驱动程序的机制。熟悉了MFC应用程序的开发方法，掌握了能开发应用程序的基本思路和技巧。

如图4，对本次实验的模块及主要思想作思维导图。

图 4 .知识点总结