

浙江大学

本科实验报告

课程名称: C++项目管理与工程实践

学 院: 计算机科学与技术学院

指导教师: 袁昕

小 组: 0errors (扫雷组)

小组成员: 于明辰、王紫晴、薛远

日 期: 2021-7-21

Our Mine Sweeping(扫雷)

1 项目背景

在本次短学期课程中，我们认识并学习了一种新型的软件开发框架，Model-View-View Model (MVVM) 模式。本项目作为小组成员学习使用 MVVM 的首次尝试，可以加深我们对 MVVM 框架的了解。经过讨论，我们决定以 windows 经典版扫雷为蓝本，使用 MVVM 模式制作经典单机游戏——“扫雷”，并在最基本的功能实现的基础上，通过多轮迭代，为其添加一些新的功能，并体现项目新特色。

2 项目需求简述

2.1 界面实现

结合经典扫雷的界面与自制、收集的素材，使用 Qt 完成游戏界面的绘制，保证界面简洁、美观，有项目特色；

为各个功能添加对应槽与窗口，规划布局，优化交互体验。

2.2 功能实现

2.2.1 基础扫雷功能

鼠标点击

用户在地图的某一方块上，可点击鼠标左键或右键，完成相应操作。

(1) 在未探索的方块上

点击鼠标左键，若该方块处为地雷，则游戏失败。否则，该方块显示出周围与它邻接的 8 个方块中的地雷数目。若该方块周围无地雷，则该方块内容为空白，并递归地进行探索，直到探索至邻接方块有地雷的方块。

点击鼠标右键，可以对该方块进行标记。根据在该方块上点击鼠标右键的次数，在未探索（无标记）、标雷（红旗）、标未知（问号）这三种模式中循环，并显示对应标记。

(2) 在已探索的方块上

若该方块周围的所有地雷都被红旗标记过，则左键/右键点击该方块时，周围非雷区的格

子均会被探索。
(此功能同样参考基础扫雷, 周围雷区都被标记证明其余格子非雷区, 相当于已被探索。此功更能方便玩家完成游戏。)

时间、雷区显示

左上角数码管显示已探索的雷区数目, 右上角数码管显示本局游戏已用时间。

胜利判定

当整个雷区中所有含地雷的方块均被标记为红旗, 且没有安全方块被错误标记时, 游戏胜利 (有不为地雷的地块被标记时, 胜利条件不触发)。

2.2.2 难易度选择

固定选项

提供了 junior、middle、senior 三种难易度的地图, 分别为 8*8, 10 booms/16*16, 40 booms/16*32, 99 booms, 玩家可以自行选择。

自定义大小

用户可以自行输入地图的行列数以及地图中的地雷数目。

2.2.3 排行榜

在用户每次完成游戏后, 储存该用户的姓名以及本次游戏时间。
在排行榜上, 可以显示用时最少的前十名的玩家姓名以及时间记录。排行榜将 junior、middle、senior 三种难易度的记录分开统计。

2.2.4 作弊模式

游戏进行中, 可以打开或关闭此模式。打开此模式时, 地图上将显示出所有地雷的位置, 便于玩家完成游戏。

3 开发与分工情况

3.1 开发计划

日期	完成工作
7 月 12 日	熟悉 MVVM 模式及相关工具, 完成分工
7 月 13 日	搭建 MVVM 项目框架, 绘制交互界面
7 月 14 日	实现左键点击, 完成基本地图绘制
7 月 15 日	实现右键点击, 基本实现游戏功能
7 月 16 日—7 月 18 日	实现难度切换 (自定义雷区)、红旗显示、创作者信息等功能
7 月 19 日—7 月 21 日	实现作弊模式、时间显示、排行榜等功能

3.2 使用工具

完成工作	使用工具
构建文件	qmake
语言使用	C++
使用软件	Qt、VScode
软件框架	MVVM
小组交流	钉钉
操作系统	windows 系统

3.3 具体分工

姓名	完成工作
于明辰（组长）	分工，管理 Git，搭建框架，Common 层与 App 层
王紫晴	管理 Git，ViewModel 层与 Model 层
薛远	管理 Git，View 层

4 项目迭代情况

4.1 第一轮迭代

完成框架搭建，绘制交互界面；
绘制固定大小地图，完成未探索区域的鼠标点击功能，完成游戏胜利与失败的判定。
实现传统扫雷游戏的大部分基本功能。

4.2 第二轮迭代

实现作弊模式，根据模式开关选择是否在地图上展示全部雷区；
完成时间的正常显示；
初步实现排行榜功能。

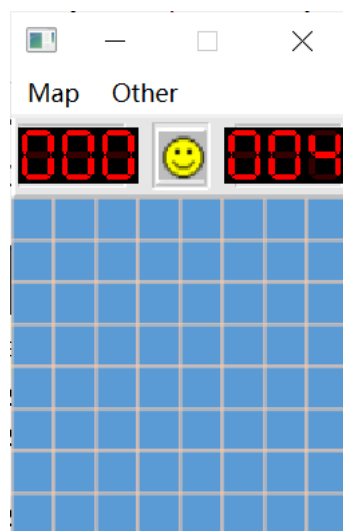
4.3 第三轮迭代

实现难度切换以及自定义地图功能；
对排行榜功能针对不同的难度模式，进行完善和扩展；
实现创作者信息显示。

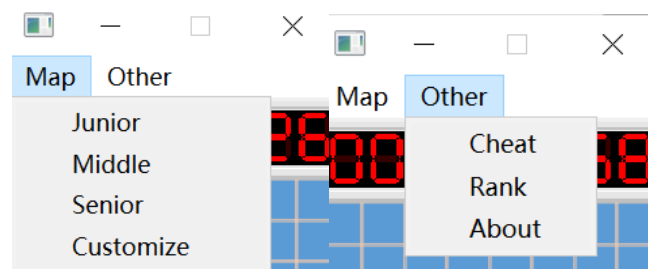
5 项目整体情况

5.1 成果展示

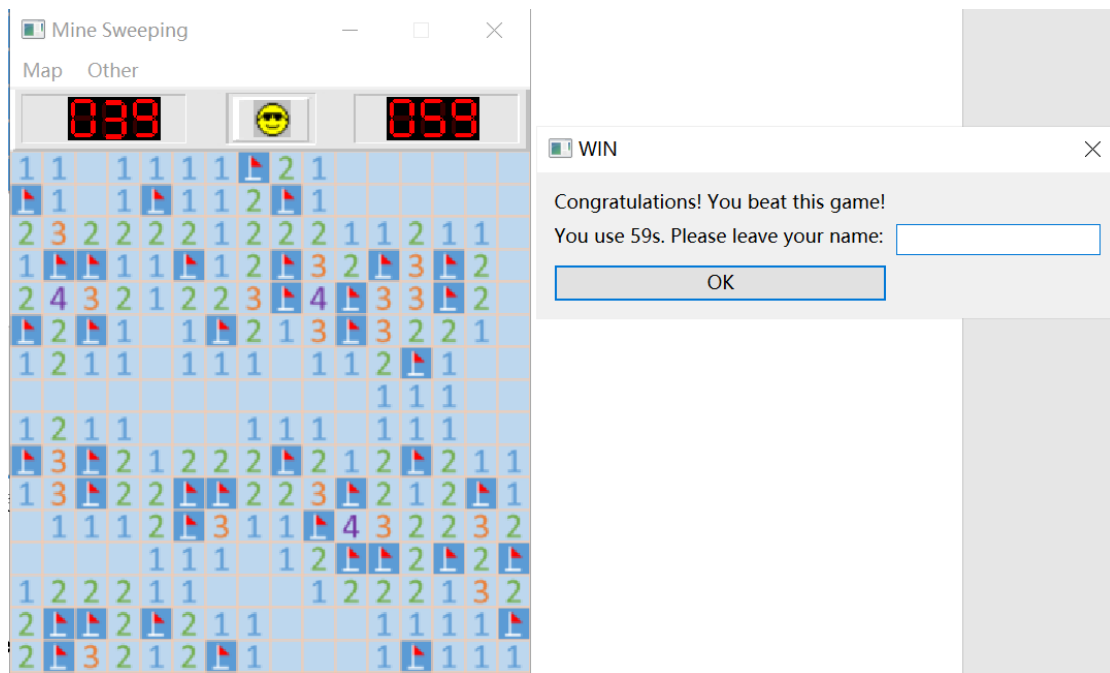
初始界面，显示默认地图：



界面功能如下：



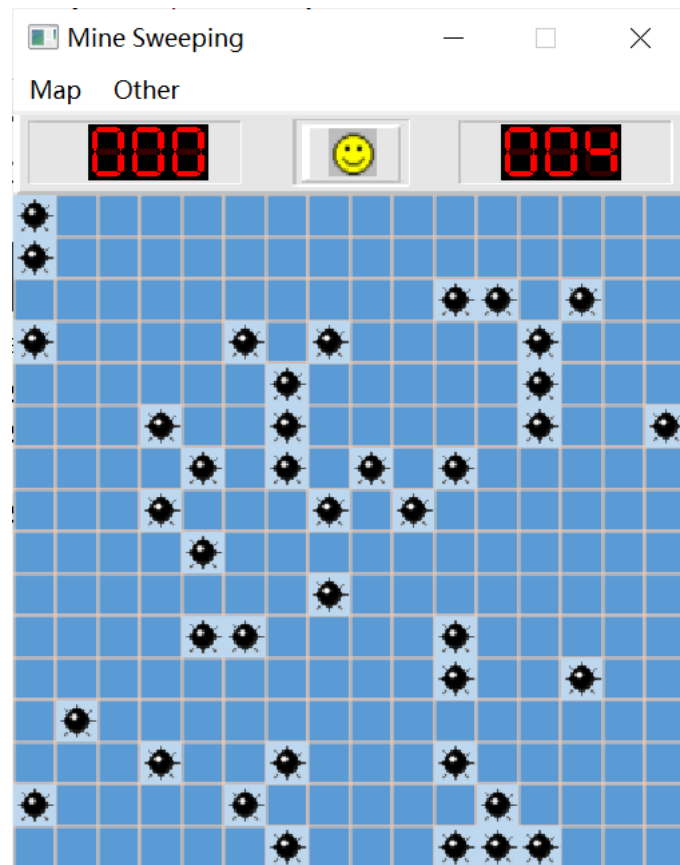
以 middle 为例，完成游戏：



踩到地雷，游戏失败：

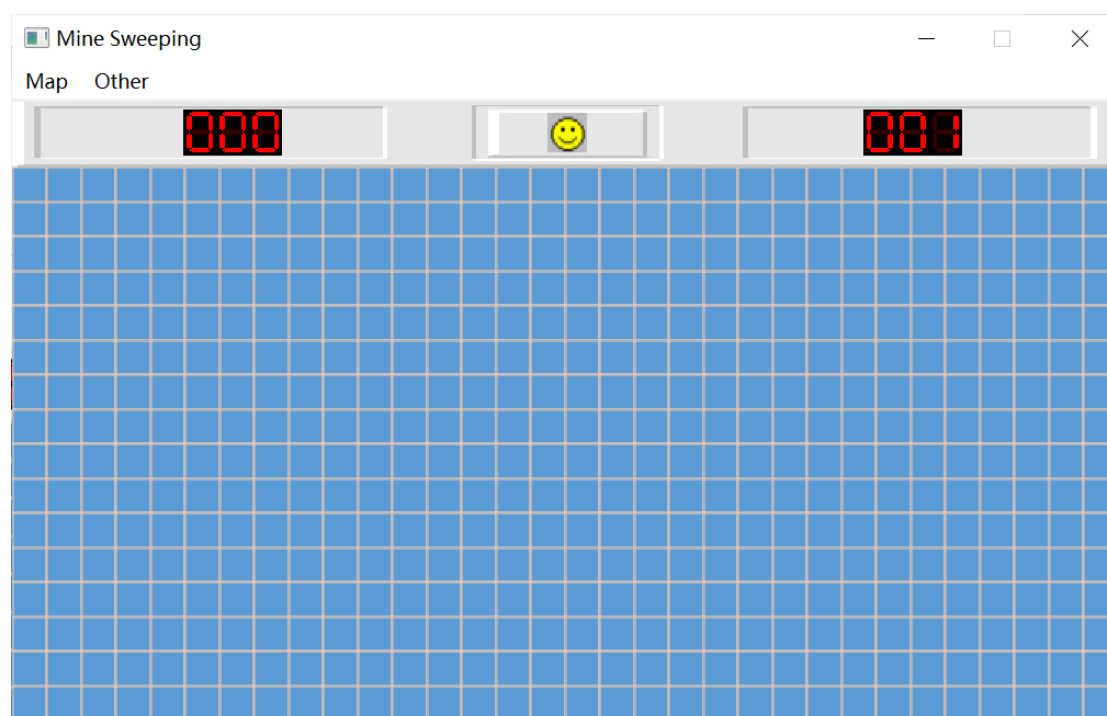
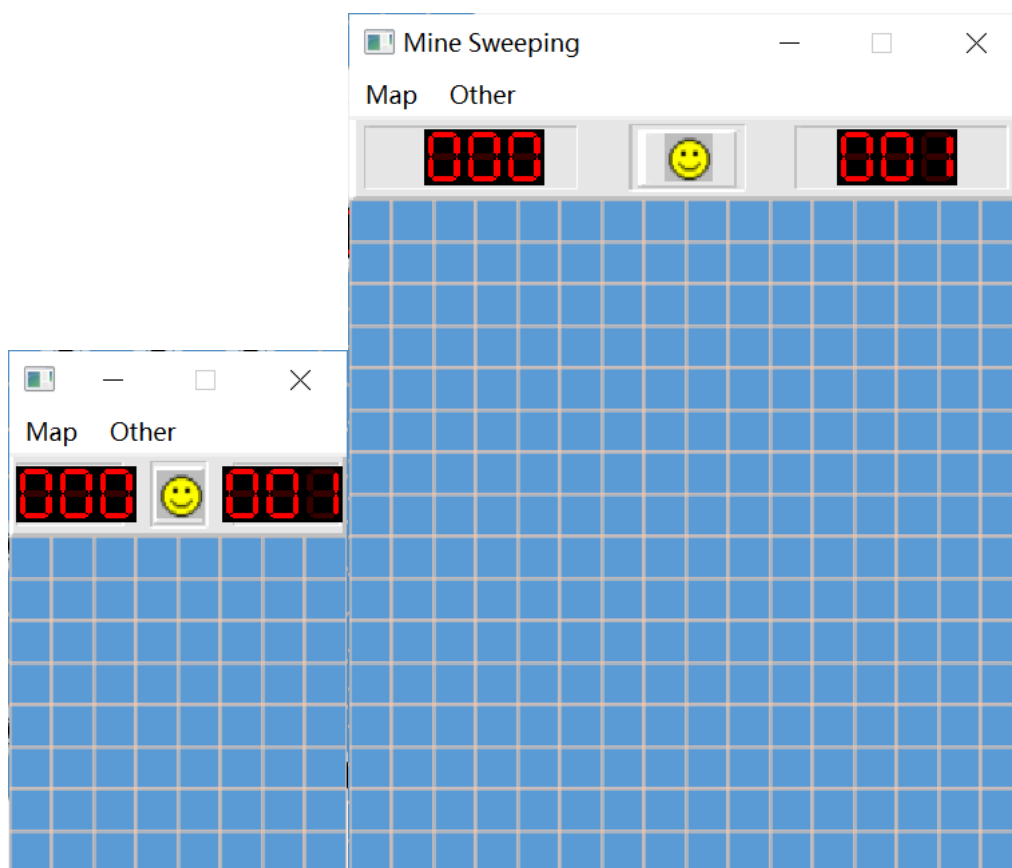


在一局新游戏打开 cheat（作弊模式）：

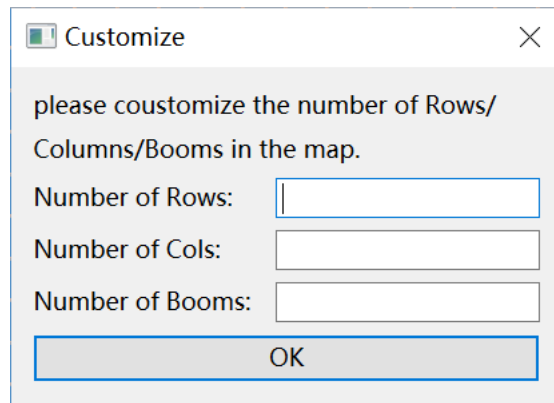


难度选择以及自定义功能：

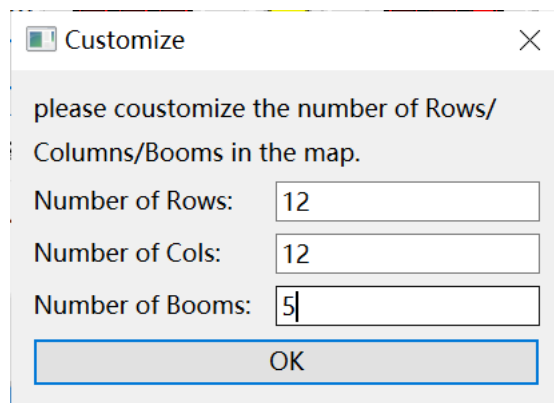
以下三个界面分别为 junior,middle 和 senior 难度。



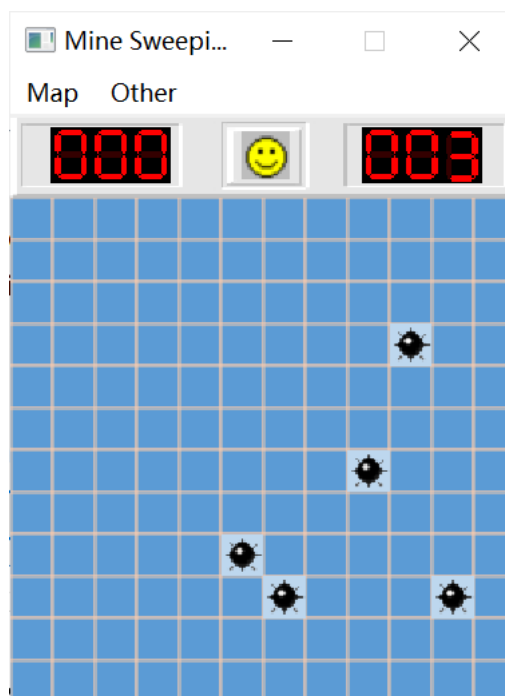
选择 customize（自定义），出现以下界面：



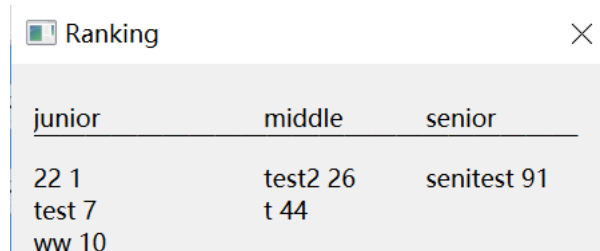
键入自定义的行数、列数和地雷数：



生成以下地图，打开 **cheat** 观测雷区，可以证明生成的地图符合输入：

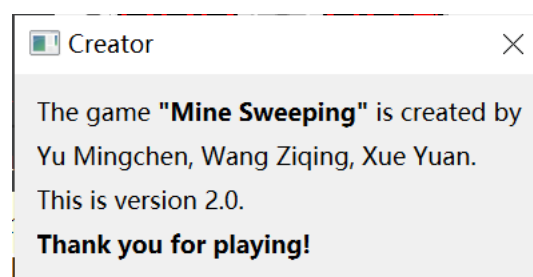


选择 **rank**，打开排行榜：

A window titled "Ranking" with a close button (X) in the top right corner. It contains a table with three columns: "junior", "middle", and "senior".

junior	middle	senior
22 1	test2 26	senitest 91
test 7	t 44	
ww 10		

可以证明三个难度模式的排行功能均能正常实现，并且都是按照时间升序排列的。
选择 **about**，查看创作者信息：

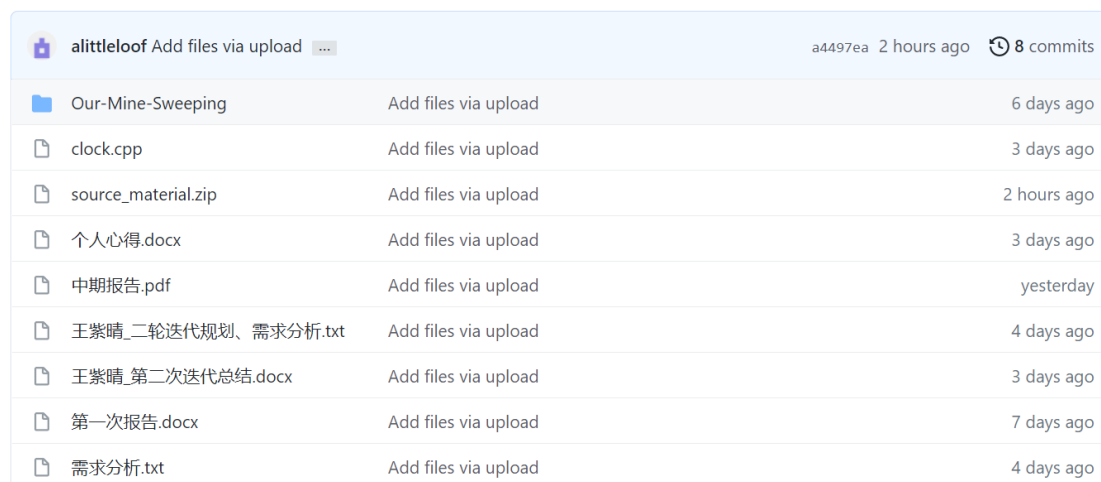
A window titled "Creator" with a close button (X) in the top right corner. It contains the following text:

The game "**Mine Sweeping**" is created by
Yu Mingchen, Wang Ziqing, Xue Yuan.
This is version 2.0.
Thank you for playing!

以上就是我们程序的主要功能。

5.2 协作情况

我们创建了一个 **github** 项目，上传工作文档，随时更新迭代情况。

A screenshot of a GitHub repository page for "alittleloof". The repository has 8 commits. The table below lists the files and folders in the repository.

File/Folder	Action	Time
alittleloof	Add files via upload	2 hours ago
Our-Mine-Sweeping	Add files via upload	6 days ago
clock.cpp	Add files via upload	3 days ago
source_material.zip	Add files via upload	2 hours ago
个人心得.docx	Add files via upload	3 days ago
中期报告.pdf	Add files via upload	yesterday
王紫晴_二轮迭代规划、需求分析.txt	Add files via upload	4 days ago
王紫晴_第二次迭代总结.docx	Add files via upload	3 days ago
第一次报告.docx	Add files via upload	7 days ago
需求分析.txt	Add files via upload	4 days ago

Contributors 3



Lucifer-ymc CSDO

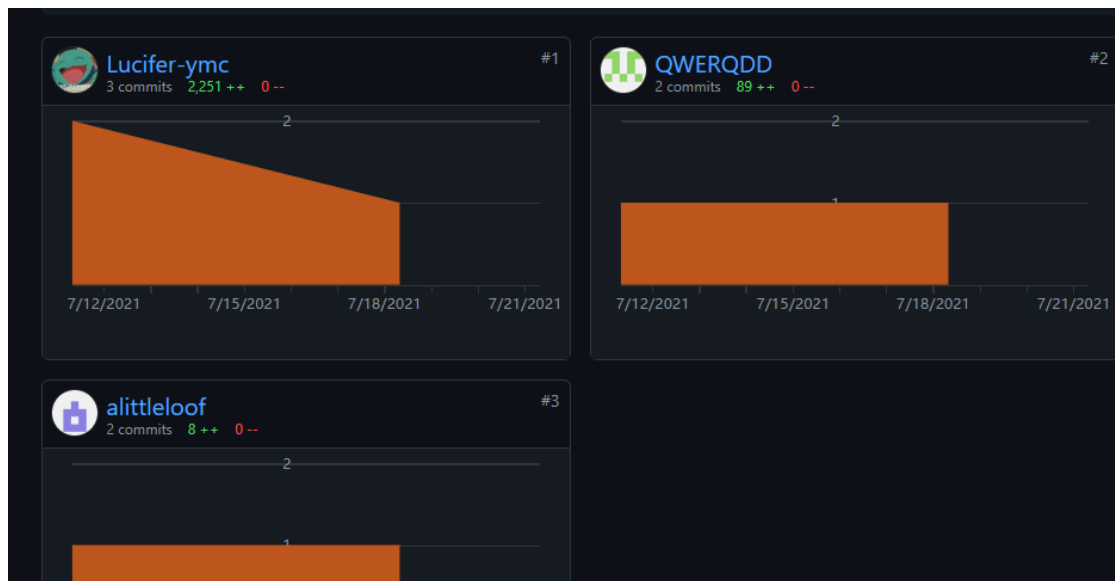


alittleloof

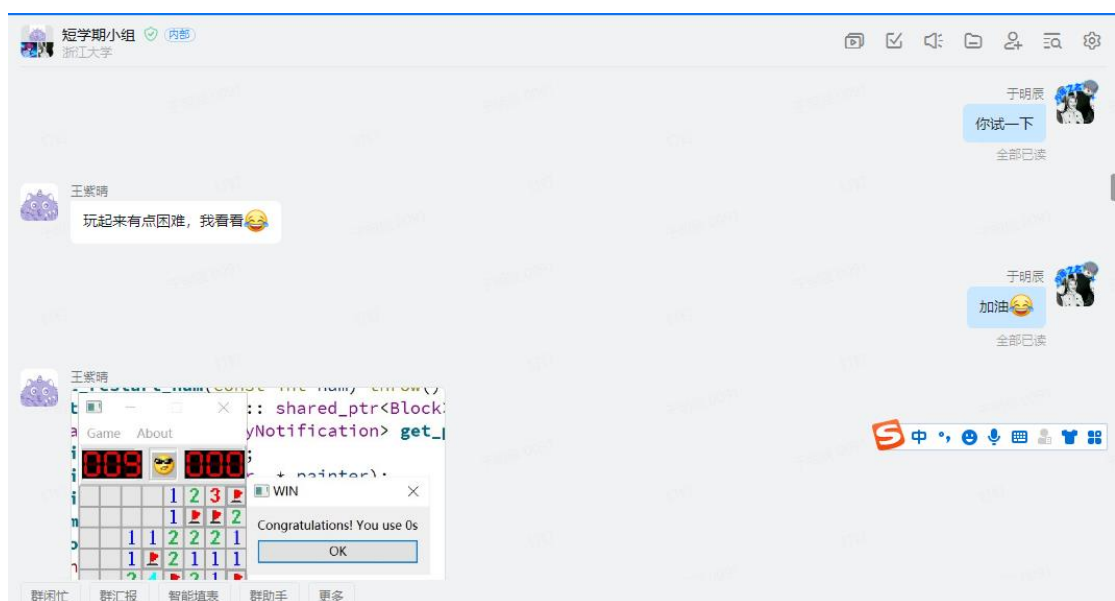


QWERQDD

贡献界面:



组内交流:



5.3 课程总结

在为期两周的 C++ 工程实践课上，我们从袁老师的讲授中学习到了很多知识，比如 MVVM 模式、怎么规范的编写代码、如何制定计划团队协作。。。总的来说，我们编写的程序虽然相比之下较为简单，但麻雀虽小五脏俱全，它很好地锻炼了我们的代码能力。在刚刚学习完 OOP 这样一门 C++ 的理论课以后，立刻试着与团队一起编写一个较有挑战性的程序，这确实是一种很好的增强 C++ 代码能力的方式，更提高了我们的自信与兴趣。同时，袁老师上课所讲的内容也使我们受益良多。现在回过头再看上学期编写的代码，很多都是不太符合规范的。在学习了这几天的内容后，感觉之后的编码之路也可以走得更加流畅了。

感谢老师几天来的辛勤教诲！

6 心得与建议

于明辰

心得：

在整个工程的三轮迭代中，我的任务主要集中在前半段——绘制雷区调试左右键等。由于之前我对 Qt Creator 不太熟悉，因此花了一天时间特地熟悉各种比如 UI 界面、.pro、三个层等的编写方法，最后总算懵懵懂懂的了解了一些。结果导致只剩一天半的时间来编写地雷块和左右键的逻辑。。。最后也刚好在老师检查成果的前一个多小时才实现了效果——也算不幸中的万幸。后面两轮的迭代我的任务相对较为轻松，也有更多的时间关注团队合作协调的工作了。整个工程总体下来，对自己分析感觉还是代码能力不够强。之前的 OOP 课期末大程在组内摸鱼，这次遇到了真刀真枪的 C++ 工程编写，代码能力不强的事实就暴露了。以后还是要加强自己编写代码的能力吧，毕竟学无止境嘛。此外，团队合作对 C++ 工程的帮助之大令我印象深刻。我能力或条件之外的一些编程部分，队友都很好的进行了完成，也让我认识到在将来编写一些比较高级的大程序时，团队合作的力量是必不可少的。

建议：

希望老师能在最开始给出一个比较具体的每轮迭代后具体交作业的时间——是这样的，我们组内有一个同学代码能力很差，然后我们一开始以为时间比较充裕，看到老师的三轮迭代大致进程就开始正常做，结果比老师要求的计划晚了一天。。。结果老师说第二天要看到相关效果的时候我们还差不少。。。结果就是那个晚上三个人很努力的在肝，最终勉强达到了效果。。。如果最开始能对老师的计划了解的能更好一些，想必应该最终效果也会更好一些吧。

王紫晴

心得:

在这门课之前,我对于 `c++` 的了解仅限于 `oop` 的一门课,也没有选过图形学相关的课, `Qt` 也是短学期前刚刚安装的。因此我们项目的实现对我来说难度还是蛮大的,一开始也感到有些迷茫,但随着时间推移,小组成员学习相关知识,并且互相交流、集思广益,问题也就一个个解决了。

一开始对于老师介绍的 `MVVM` 其实没有什么体会,等到实际合作的时候才慢慢感受到它比起从前大程合作时所谓“一人一部分”更加优越的地方。解耦合的实现,对于多人合作可以说是非常友好。

通过这次的项目,我学到了新的软件框架 `MVVM` 模式,对 `Qt` 的使用也远比以前熟练,从画界面中体会到了很多乐趣。总而言之,短学期的课程让我感到受益匪浅,相信从中学到的这些知识在以后的学习中也会使用到。

建议:

提出一个有些伸手党的建议,希望老师能提供一一些 `Qt` 相关的资料,对于零基础选手来说,短时间内弄清楚用 `Qt` 绘制 `mainwindow` 界面的相关函数有一定困难。

薛远

心得:

我对于本次 `c++` 大程序贡献十分有限,在此非常感谢我的组长和另一位组员。

我配置环境遇到非常大的困难,相继解决了 `MSB8020` 和一些诸如 `any` 类代码无法通过的问题。

虽然我在代码方面贡献不足,但是这次 `app` 开发的促使我去学习 `qt`。我因此在 `b` 站上完整的观看了 `qt` 教程规范,理解了 `slot` 槽的用途,一些具体指令的规范。

最后,我对时间实现有了一定的认识,学习了 `qt` 时间函数的返回功能和随即功能。

这次短学期,中间穿插着 `pre`、搬宿舍以及今天出了车祸在警察调节下才解决的一些事情,拖了队友很大的后腿,再次表示歉意。