



哈爾濱工業大學(深圳)
Harbin Institute of Technology Shenzhen

《创新训练课 B》结题报告

学院: 机电工程与自动化学院

题目: 模拟地铁自动售票系统

班级: 自动化与电气工程 6 班、8 班

姓名: _____

学号: 2023312605、2023312821

教师: _____

上交日期: 2024. 12. 13

修订历史记录

日期	版本	说明	作者
2024. 7. 15	V1. 0	1. 实现基本功能 2. 搭建完程序框架	
2024. 7. 18	V1. 1	1. 修复人数为 0 时也能购票的 bug 2. 修复找零部分的 bug	
2024. 12. 11	V2. 0	修复支付金额上限不足的 bug	

目 录

1 引言	4
1.1 编写目的.....	4
1.2 背景.....	4
1.3 定义.....	4
1.4 参考资料.....	4
2 任务概述.....	4
3 需求分析.....	2
3.1 用户需求分析.....	2
3.2 运行环境.....	2
4 功能及操作介绍.....	2
4.1 操作.....	2
4.2 功能.....	3
5 系统设计.....	3
5.1 总体架构设计.....	3
5.2 模块分析与设计.....	3
5.3 软件结构（流程图）.....	4
6 调试与测试.....	5
6.1 调试过程.....	5
6.2 测试结果.....	5
7 编程中遇到的问题.....	7
7.1 问题 1.....	7
7.2 问题 2.....	7
7.3 问题 3.....	7
8 分析总结与心得体会.....	7

1 引言

1.1 编写目的

提高购票效率：通过自动化系统减少购票时间，特别是在高峰期，避免购票窗口排长队的现象，提高整体运营效率。

减少人为错误：自动化系统减少了售票过程中可能出现的找错钱、卖错票等人为错误，提高购票准确性。

数据管理和分析：系统能够实时记录和统计票务信息，便于管理部门分析乘客流量和收入情况，支持运营决策。

1.2 背景

随着城市地铁线路的不断扩展和客流量的增加，传统的人工售票方式已经无法满足需求。自动售票系统可以实现快速购票、减少排队时间，提高地铁运营效率和乘客满意度。

人工售票需要大量的售票员，不仅增加了人力成本，还可能存在售票错误等问题。自动售票系统可以降低运营成本，提高售票效率和准确性。

城市的快速发展需要高效、便捷的交通方式。地铁作为一种大运量、高效率的交通工具，自动售票系统的开发可以更好地适应城市发展的需求，提高城市交通运输的效率和质量。

1.3 定义

模拟地铁自动售票系统是一种软件系统，它模拟了真实地铁自动售票系统的功能和操作流程。该系统通常包括线路选择、站点选择、车票类型选择、支付方式选择、车票购买、退票、查询余额、充值等功能。

1.4 参考资料

B 站 Python Qt 教程

《Python 学习手册（第四版）》

2 任务概述

研发一个模拟地铁自动售票系统，模拟实现现实中地铁自动售票系统的一些功能。本项目中，模拟地铁自动售票系统包括三个子项目，分别是“地图选点”“查看方案”“找零”。

3 需求分析

3.1 用户需求分析

地铁自动售票系统的用户需求主要包括以下几点：

1. 简单易用：系统界面简洁明了，操作流程简单易懂，方便用户快速购票。
2. 实时信息查询：查看地图、线路、最小换乘线路和最少站点线路等。

3.2 运行环境

Win11 等

4 功能及操作介绍

4.1 操作

先点击“开始购票”。

(1) 地图选点：

1. 系统会显示石家庄地铁的地图的一、二、三号线的信息，包括沿途的车站、换乘点等。
2. 地图的缩放和移动可以通过**鼠标的滚轮和左键拖拽**来实现，也可以通过右下角的**按钮控制地图的放大、缩小以及重置**。
3. 选择站点时先点击起点站或终点站的输入框，然后可以通过**直接在地图上选点或输入**的方式选站，输入时可以通过键盘上下箭头以及 Tab 键进行自动填充。
4. 界面右下方可以预览站点的中英文名和所在线路

(2) 查看方案：

1. 系统展示**最小换乘线路**（可能不止一条）**和最少站点线路**，帮助用户更高效地规划行程。其中，线路中会显示起点、终点站、换乘站、途经站点数、每条线对应的方向（只显示途经站点数而不显示途经站点名称，在途经站点数较大的界面更清晰）。
2. 支付界面同时给出一个或多个最少换乘的方案，以及一个**经过站点数最少**的方案。
3. 计算阶梯票价：
<1>确认是否往返，若往返则需支付金额加倍。
<2>输入人数（1-50 之间的整数）。
<3>输入支付金额（1-1500 之间的整数，且只有付款金额不低于所需金额时方可结算），点击确定，进入下一界面。

其中，支付界面只会给出一个价格，这个价格是按照经过的站点数最少的情

况来计算的。

【注：本程序的价格计算公式为： $2 + [\text{同一线路上起始站与终点站间的站点数} / 5]$ （向下取整），而现实中为按最短里程分段计价，二者有所不同】

（3）找零：显示找零的金额（找零的金额=支付金额-需付金额），并可返回主界面。

4.2 功能描述

地图选点

地图显示：展示地图信息，帮助用户直观地了解站点、线路等地理分布。

站点查询：用户可以通过输入站点名称来查询特定的站点信息。

线路查看：在查询到站点信息后，用户还可以进一步查看该站点所在的线路，了解站点与其他站点之间的连接关系。

查看方案

查看最小换乘线路和最少站点线路：根据用户键入的起始站和终点站，查看最小换乘线路（可能不止一条）和最少站点线路。

计算阶梯票价：

- (1) 确认是否往返，若往返则需支付金额加倍。
- (2) 输入人数。
- (3) 输入支付金额（只有付款金额不低于所需金额时方可结算），点击确定，进入下一界面。

找零

找零的金额=支付金额-需付金额。

5 系统设计

5.1 总体架构设计

采用分层式架构，系统分为以下三个层次：表示层（也称为 UI 层）、应用层（也称为服务层）、业务逻辑层（也称为领域层）。

5.2 模块分析与设计

模拟地铁自动售票系统分为三个模块，分别是“查询地图”“购票”“支付”。各模块功能如图。



图 1 模块示意图

5.3 软件结构（流程图）



图 2 软件结构流程

6 调试与测试

6.1 调试过程

(1) 问题：发现有时换乘两次比换乘一次便宜，如从“博物院”到“孙村”。也即“广度优先搜索”(BFS)的方法找到的是最少换乘的方案，不是最少站的方案。

解决：采用计算最少站即最小金额的算法。

(2) 问题：人数为 0 时也能购票，且人数上限和购票金额上限都为 99，若金额超过 99 元则无法支付。

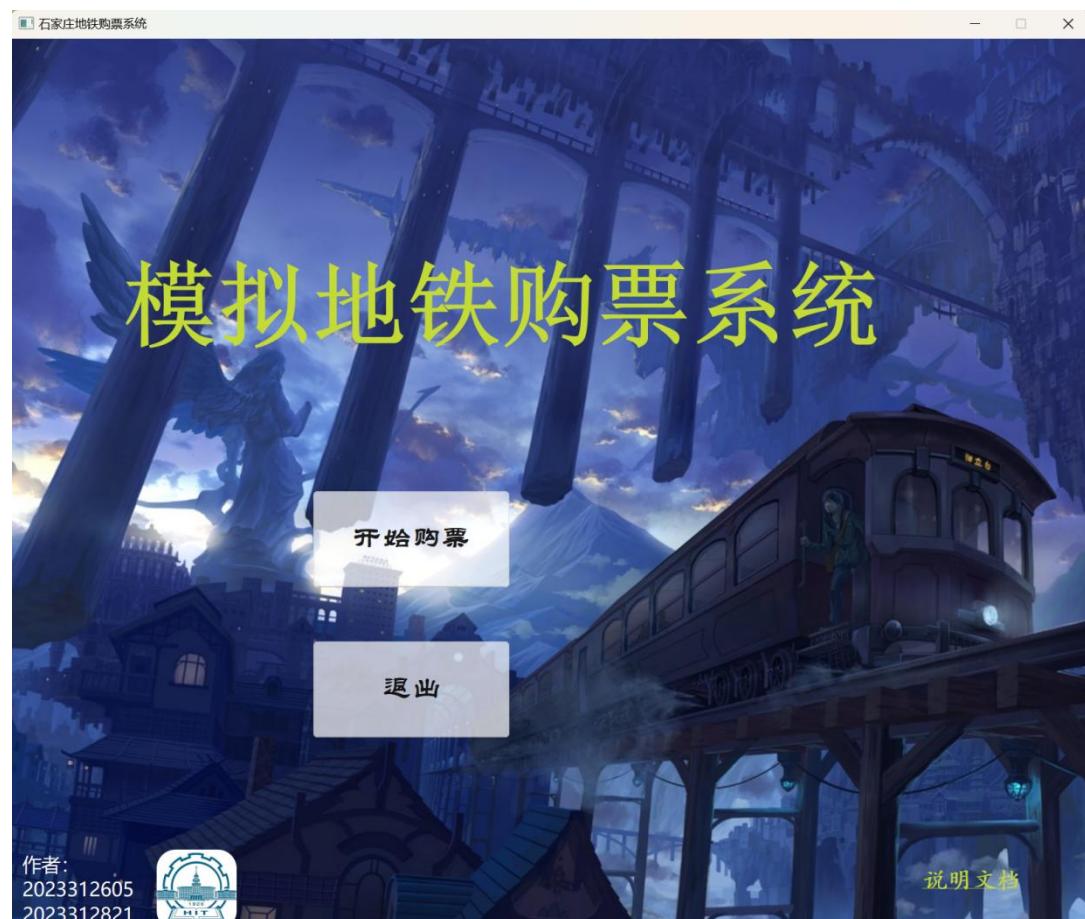
解决：输入的人数限制为 1-50 之间的整数，输入支付金额限制为 1-1500 之间的整数，且只有付款金额不低于所需金额时方可结算。

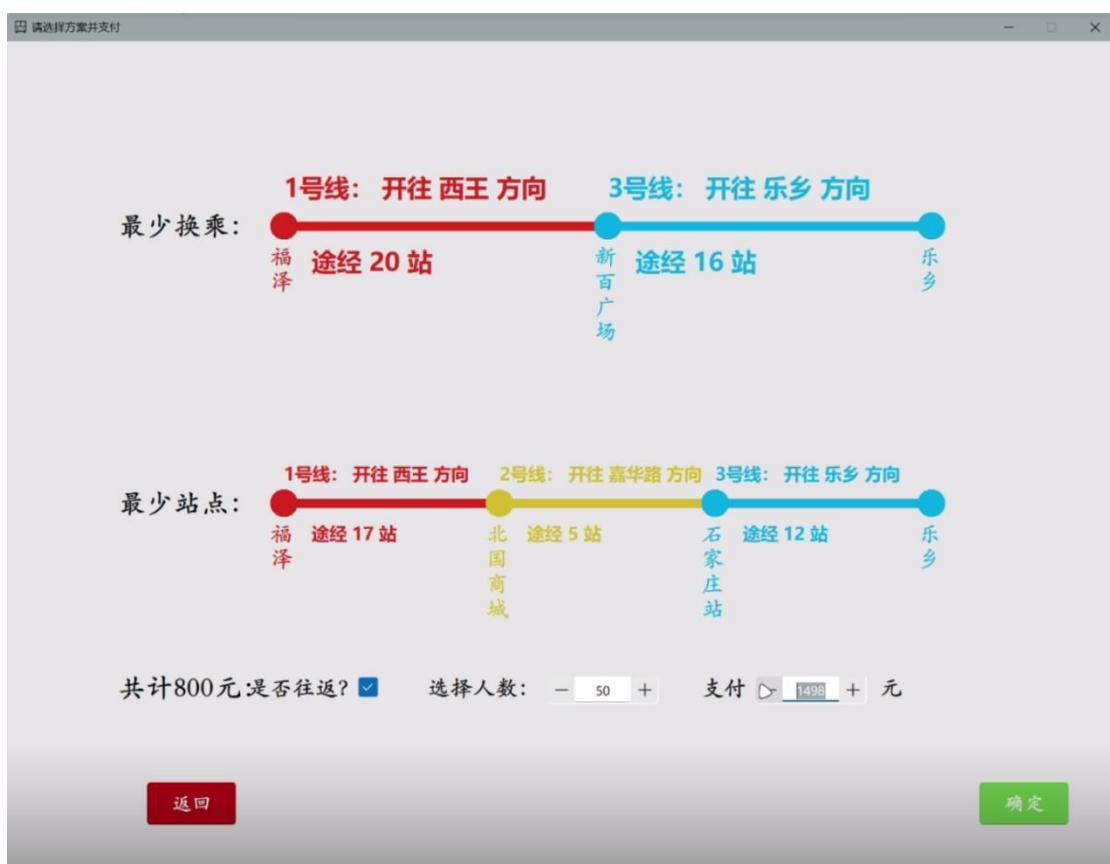
(3) 问题：第一次是有找零，第二次刚好不找零的时候，而且两次之间不退出系统时，显示的找零钱数有时候是第一次的。

解决：优化了找零部分的代码，不找零时不会出现文字提示。

6.2 测试结果

程序正常运行示例：







7 编程中遇到的问题

7.1 问题 1

开题报告中的有些部分难以实现。因此在实际过程中采取代替方案。

7.2 问题 2

采用各站点间实际的金额较为烦琐。因此本程序的价格计算公式为： $2 + [\text{同一线路上起始站与终点站间的站点数} / 5]$ （向下取整），而现实中为按最短里程分段计价，二者有所不同。

7.3 问题 3

地图选站界面按钮本身不会缩放，因此在地图缩放时会偏离地图中的站点。

7.3 问题 4

有些组件的堆叠可能出现次序上的问题。并且在系统深色模式下有些组件的显示会出现问题。

8 分析总结与心得体会

在本项目中，我们选用 QT 框架，通过 C++ 来模拟一个地铁自动售票系统。这个过程既考验了我们的编程技能，又是一次团队协作与问题解决的实战演练。

在合作过程中，我们深刻体会到了团队协作的重要性。每当遇到问题时，我们会一起讨论，共同寻找解决方案。这种团队精神提高了我们的工作效率。

此外，这次实践也让我们对 QT 框架有了更深入的了解。通过实际操作，我们掌握了 QT 的基本使用方法，学会了如何设计并开发一个功能完善的图形用户界面。这不仅巩固了我们的编程知识，更拓宽了我们的编程视野。

未来的程序还可以向以下几个地方优化：可以优化为支持**多种支付方式**：无现金交易，用手机通过微信或支付宝扫码支付，迎合乘客的支付习惯；可以加上**背景音乐**，且用户可以控制播放与暂停；可以加入**音效或语音播报**。

总之，这次双人合作编程项目是一次充满挑战与收获的经历。它不仅提升了我们的编程技能，更让我们学会了如何在团队中更好工作。

9 项目分工

zzk: ui 设计与框架构建

xlh: 调试改进与材料撰写