

1 软件开发环境

1. 操作系统：Windows 10 Professional x64
2. 开发语言：C++
3. 集成开发环境（IDE）：Visual Studio Community 2019
4. 图形界面开发库：Qt 5.14.0
5. 版本控制：Git for Windows v2.25.1

2 总体结构

本系统采用面向对象的程序设计思想，总体结构可分为“交互层”“控制层”和“算法层”。

其中交互层负责响应用户操作，向控制层发送请求，并将控制层输出的信息通过 GUI 反馈给用户；控制层负责系统的整体运行，以及数据维护/处理，日志记录等；算法层负责根据旅客请求生成相应的旅行方案，并交付控制层执行。



3 模块划分

3.1 控制层

控制层负责系统的整体运行，以及数据处理，日志记录等。控制层可以划分为以下四个模块：

1. 系统初始化模块

启动日志记录；导入城市信息和车次信息；启动系统时钟；启动相关线程。

2. 旅行实例创建模块

响应旅客要求，将旅客的要求封装成算法层可以接收的算法输入；创建旅客旅行实例；并记录旅行实例的提交时间。

3. 旅行模拟模块

旅行模拟模块根据旅行方案对旅行过程进行时序模拟，并实时输出到交互层供用户查看。

4. 日志记录模块

日志记录模块记录旅客状态的变化和系统状态的变化。输出到日志文件，也可交付到交互层供用户查阅。

3.2 算法层

算法层负责根据旅客请求生成相应的旅行方案，并交付控制层执行，是该系统中最核心的部分。算法层自成一个模块。

具体模块的算法设计详见《各模块设计说明》。

【注】部分模块由子模块构成，本文仅给出系统的总体说明，进一步细分的模块结构及具体实现未作介绍，详情可查看“各模块设计说明”文档。

3.3 交互层

交互层负责响应用户操作，向控制层发送请求，并将控制层流出的信息通过 GUI 反馈给用户，交互层可以划分为以下模块：

1. 旅客信息录入模块

旅客信息录入模块负责录入旅客旅行的起点、终点和策略，将这些基本信息交付控制层进行封装。

2. 旅行信息输出模块

旅行信息输出模块负责将控制层模拟的旅行过程进行展示，包括旅行方案、旅客的实时位置（地图），以及旅行进度等相关信息。

3. 旅客切换模块

旅行信息输出模块负责切换旅客，展示不同旅客的旅行信息。

4. 用户界面刷新模块

用户界面刷新模块负责更新界面控件的可用性以及界面显示信息，包括但不限于系统时钟、所选旅客信息和系统信息。