

1 数据结构说明

本部分将列举旅行模拟查询系统中使用到的所有基本数据结构，部分Python语言内置的数据结构如存放用户名的字符串、标记用户是否勾选“不限”的整数型变量以及简单使用了以下数据结构的一些其他基本结构本文不作详细说明，详情可参考程序源代码。

本部分除了给出数据结构的介绍以外，还针对各数据结构给出了示例。示例中部分为程序代码中摘出的实际代码，另一部分为了方便说明而书写的伪代码。

1.1 旅客信息

- 数据结构：字典，列表；
- 存储内容：存储各个旅客的相关信息。其中包括出发城市、目的城市、时间限制、出行方案、出发时间、旅客编号、是否抵达某个城市(in_trans)、目前位于旅行方案的第几段(route_index)、是否抵达目的地(arrived)、当前所在城市、方案风险。

【示例】

```
traveller={ 'id': 0, 'departure': '北京', 'destination': '兰州', 'route':  
[{'departure': '北京', 'destination': '天津', 'way': '汽车', 'id': 'A5215',  
'depart_time': 10, 'arrive_time': 12, 'time_interval': 10, 'trans_time': 2},  
{ 'departure': '天津', 'destination': '沈阳', 'way': '汽车', 'id': 'A3699',  
'depart_time': 12, 'arrive_time': 15, 'time_interval': 12, 'trans_time': 3},  
{ 'departure': '沈阳', 'destination': '哈尔滨', 'way': '汽车', 'id': 'A1570',  
'depart_time': 10, 'arrive_time': 19, 'time_interval': 34, 'trans_time': 9},  
{ 'departure': '哈尔滨', 'destination': '兰州', 'way': '火车', 'id': 'T83',  
'depart_time': 21, 'arrive_time': 7, 'time_interval': 45, 'trans_time': 10}],  
'in_trans': 0, 'route_index': 0, 'time_limit': 0, 'arrived': 0, 'present_city':  
'', 'start_time': 0, 'risk': 49.0, 'time': 55}
```

1.2 城际交通信息图

- 数据结构：字典；
- 存储内容：构成了城市之间的邻接图，包括每个城市的所有交通班次和权重

【示例】

```
graph: { '北京': { 'transportation': [{ 'departure': '北京', 'destination': '上海',  
'way': '飞机', 'id': 'CA957', 'depart_time': 9, 'arrive_time': 11}, { 'departure':  
'北京', 'destination': '上海', 'way': '火车', 'id': 'T103', 'depart_time': 20,  
'arrive_time': 9}, { 'departure': '北京', 'destination': '广州', 'way': '火车',  
'id': 'T15', 'depart_time': 18, 'arrive_time': 16}, { 'departure': '北京',  
'destination': '哈尔滨', 'way': '火车', 'id': 'T87', 'depart_time': 22,  
'arrive_time': 9}, { 'departure': '北京', 'destination': '沈阳', 'way': '火车',  
'id': 'T91', 'depart_time': 7, 'arrive_time': 16}, { 'departure': '北京',  
'destination': '沈阳', 'way': '汽车', 'id': 'A2547', 'depart_time': 20,  
'arrive_time': 12}...]}}
```

1.3 时刻表

- 数据结构：列表、字典；
- 存储内容：程序获取的所有班次的详细安排，列表中所存储的为某一班次的出发城市、目的城市、交通工具、班次、出发时间、抵达时间，时刻表列表由上述字典构成。

【示例】

```
Schedule=[{"出发地": '北京', "目的地": '上海', "交通工具": '飞机', "班次": 'xx124', "出发时间": xxx, "到达时间": xxx}, {"出发地": '上海', "目的地": '海口', "交通工具": '飞机', "班次": 'xx141', "出发时间": xxx, "到达时间": xxx}]
```

1.4 用户旅行任务

- 数据结构：字典；
- 存储内容：通过用户界面获取的用户输入的旅行任务。其中包括出发城市、目的城市、时间限制、出行方案、出发时间、旅客编号。

【示例】

```
mission_dict = {'departure': '北京', 'destination': '上海', 'time_limit': 100, 'plan': 0, 'start_time': xxx, 'traveller_id': 5}
```

1.5 用户路线方案

- 数据结构：列表，字典；
- 存储内容：存储用户得到的最佳路线的信息，以列表的形式存储，列表的每一项为途径城市及班次号的相关信息。列表的每一项包含该班次的出发地、目的地、交通工具、班次号、出发时间、抵达时间、在出发地等待时间、搭乘交通工具的时间。

【示例】

```
route=[{'departure': '北京', 'destination': '天津', 'way': '汽车', 'id': 'A5215', 'depart_time': 10, 'arrive_time': 12, 'time_interval': 10, 'trans_time': 2}, {'departure': '天津', 'destination': '沈阳', 'way': '汽车', 'id': 'A3699', 'depart_time': 12, 'arrive_time': 15, 'time_interval': 12, 'trans_time': 3}]
```

2 数据字典

此部分主要对本系统中所使用到的关键变量、函数的名称、作用以及数据结构等进行介绍，而中间计算过程所使用的临时变量以及一些非关键变量等本部分不作详细说明。

2.1 关键变量

变量名	类型	作用
init_time	datetime.datetime	常量，记录程序的初始时间
city_id	字典	用于记录每个城市对应的id
time_gap	整型	常量，用于推进时间
city_graph	字典	城际交通信息图
schedule	列表	所有交通班次的详细安排
traveller_list	列表	所有旅客的信息
map_on	整型	标记地图模块是否打开
time_thread	Threading.Thread	时间进程，推进时间
mission_dict	字典	记录每个输入任务信息
present_time	整型	当前时间与初始时间的差
update_traveller_text_signal	pyqtSignal	更新地图模块文本框的信号
draw_map_signal	pyqtSignal	打开地图模块的信号
time_update_signal	pyqtSignal	更新时间的信号

2.2 关键函数

- 主窗口

函数名	作用
initSlot	初始化pyqt信号和槽
__init__	初始化主窗口相关信息
__del__	退出主窗口的处理函数
stop_timer	关闭计时器
start_timer	打开计时器
setupUi、retranslateUi	初始化GUI
time_update	更新时间及相关模块
change_dep_combo	修改任务提交按钮状态
get_mission	生成任务并计算方案
get_route	计算最佳路线
print_route	方案展示模块
update_traveller_state	每小时更新旅客的状态
open_map	打开地图

- 地图模块

函数名	作用
initSlot	初始化pyqt信号和槽
__init__	初始化地图模块相关信息
update_traveller	更新旅客信息
setupUi、retranslateUi	初始化GUI

- 时间模块

函数名	作用
run	时间线程推进时间的函数
__init__	初始化时间模块相关信息
stop_timer	停止推进时间
start_timer	开始推进时间