Project 3

中大游览

班级：教务三班

彭高15331253

刘俊良 15331215

刘钟涛 15331222

## 「题目要求」

## 「数据结构与算法」

数据结构的选择上，我们采用有向图来模拟整个地图。具体实现中，考虑到整个地图的后期的延展性（插入删除的效率），我们使用了邻接链表实现方法。

整个程序分为三个类：CampusMap，Guide，Sight。CampusMap用于表示单独的一个地图，Guide用来表示导游系统（可以选择校区），Sight用来表示具体的景点，存储景点描述等信息。在CampusMap中还定义了两个结构体vertex和edgeVertex，分别表示邻接链表中的节点和边。

为了使地图信息静态化，我们使用了一个json文件来存储当前的地图信息，每次运行C++程序的时候利用外部引入的Json类解析json文件，读入地图信息

为了提升编码效率，我们采用了C++11中许多新特性，比如利用智能指针shared\_ptr连接邻接链表从而免去了普通指针的麻烦，再比如利用C++11中的自动类型判定decltype，auto，提高了编码效率和程序稳健性

图1 Person类、CampusMap类、Sight类UML

#### CampusMap类

1. 成员spots存储当前地图中所有的节点（景点）
2. 成员campusName表示当前地图的名字
3. 方法findSight用于通过景点名字获得景点在spots中的迭代器
4. 方法addVertex用于向spots中添加景点
5. 方法findShortestPath（使用Dijkstra算法）寻找两个景点之间的最短距离
6. 方法extractMin为方法findShortestPath的辅助函数
7. 方法findDistance用于寻找两个景点之间的直线距离（如果没有直接连接的路线则返回无穷大）

#### Guide类

1. 成员SYSUguide表示导游，存储所有的地图信息（东校区、南校区、北校区、珠海校区）
2. 成员 thisMap表示当前Guide在导览的地图
3. 方法findCampus用于通过地图名称寻找地图
4. 方法showSight用于展示当前地图中名字为name的景点信息
5. 方法start呈现表格选单

#### Sight类

1. 成员sightName表示景点名称
2. 成员intro表示景点信息

#### 「数据测试、结果及分析」

由于本次题目地图信息有限，所以我们测试了地图中所有点之间的最短距离。

最终的结果如下













