**赛题场景**：

机票预定过程中往往需要多次搜索，搜索请求规模大（查订比1000:1）

性能要求：

高性能（后台引擎响应延迟低，并发高）、价格低、命中率高的搜素服务，必须返回票价最低的方案

难度评估：

企业评估认为赛题较难，非功能部分作为重要加分项

任务要求：

* B/S结构前后端，前端浏览器实现搜索请求输入以及展示（前端技术无任何限制）
* 后台服务为前端web提供实时机票搜索推荐服务，需要考虑非功能性要求
* 前后端web服务可分开，即前端web服务可调用一个搜索的web服务

解释说明：

* 代理人：代理人表示不同的出票平台，比如携程、同程、航空公司官网等，身份不同票价也有差异。举例中一个结果有两个代理人，表示这两个代理人的任意一个都可以出售该结果中的机票。
* 不考虑为用户中转航班使得从A到B，忽略中间过程
* 只有用户自己选择转机的情况且满足各种中转时间间隔要求（如转机间隔一天或一个月），需要有报错信息（如请求了去年的时间），需要考虑飞机衔接时间的可行性（如第二段的飞机不能比第一段起飞早）

功能要求（建议阅读赛题中的例子）：

* 低价行程推荐（最基本功能）
  + 支持若干航班（不超过8个）联程推荐功能（一个人同时询问连续多个航班，例如环球飞，如8段航班返回8个结果），限定旅客人数为1，出票代理人为1（只能从一个代理人那儿购买），返回若干最低价行程结果
  + 每个最低价行程结果包含行程中的所有具体航班、舱位、总票价等信息
* 多代理人行程推荐：支持输入多个出票代理人（最多20个），一个价格多个平台，要求每个低价行程结果附带可出票的所有代理人信息。
  + 同一个航班组合，可能出现不同代理人的出票金额不同的情况，此时相同票价金额视为一个低价行程结果
  + 每个低价行程结果中的所有航班必须由相同代理人出票，不能切割成不同代理人出票
* 多旅客行程推荐：在多代理人低价行程推荐的需求上，增加多个旅客（最多8个）同时搜索的功能
  + 每个低价行程结果中所有旅客同机乘坐；
  + 每个低价行程结果中所有旅客在某一个代理人处一起出票，也就是说旅客不能拆分到不同代理人处出票；
  + 旅客数量不能超过该航班余座数值；
* 航班余座数据实时更新：航班余座数值初始值由选手自主随机生成，余座数值变更由外部模拟系统主动发送变更报文触发，更新频率较低（比如一个航班的平均更新频率大概为每天1次。一般规律，一个月内即将起飞的航班比一个月后的更新频率更高，以此类推)。

算法要求/非功能性要求：

* 依据航班、运价、余座等三类信息（运价参考里程，需要考虑三者的逻辑关系，重点）
* 完成搜索引擎，包括数据预处理或索引、最佳路径选择等，避免使用穷举法
* 最好考虑数据合理规模（参考国内一年的航班数量，非重点）
* 在一个进程内完成搜索
* 当余座频繁更新时如何数据访问（余座信息必须实时更新，搜索不得终止，其他数据可认为不更新，余票数字可能会变大）
* 同一台服务器内所有搜索服务进程/线程共享唯一一份航班、运价、航班余座等数据，对应索引也要共享
* 数据库不推荐第三方（赛题数据库是只读类型，满足非功能性要求第三方库无法满足）
* 部署服务器必须可以公共访问以便企业测试
* 云服务器考虑主流CPU性能
* 搜索是只读的，没有出票环节

赛题建议：了解如Linux系统调用，HTTP协议，数据库原理

数据格式：

数据长度必须格式相同，时间精确到分钟











