**方案设计与论证**

**总体分析**

这个系统需要攻克的主要难点有三个，分别是图形界面、高效算法和数据结构设计、文件的存储方式。我们不想用命令行来实现这个程序，于是就需要用到图形界面，需要将它设计的让用户觉得友好。同时高效的算法和数据结构可以保证用户永远能得到快速的响应，使用时不需要过多的等待，程序也不需要耗费过大的内存资源。为了保证每次初始化时不需要将已有的数据重新导入一遍，所以我们需要借助硬盘的存储。同时程序也不能只靠用户一个一个输入来加入数据，也需要能够支持批量的从外部文件导入数据，亦或是将数据导出到外部文件供其它应用程序使用。

**基本思路**

当运行程序时进行初始化，将保存在硬盘里面的数据反序列化回内存中。同时展示图形界面，后台程序等待图形界面的调用。

图形界面用Swing构建，通过调用方法与后台模块交互。图形界面端的模块需要处理调用方法返回的数据，并将他们妥当的显示在图形界面上。

后台需要提供一个完整的方法列表，每个功能均至少有方法与之对应对应。返回的数据需包含前台需要用到的，并且便于前台展示。

后台在实现相应功能时可以调用各种方法，这些方法主要包括查询、更改、添加。这些方法均需要高效的实现，并且与这个程序独特的数据结构对应。

导入的步骤中包括解析csv，并转换为循环插入。导出即为查询并写入csv文件。在程序运行结束时需要将内存中构建好的数据结构序列化到硬盘中，方便下次启动的时候读取。

**遇到的问题和解决方法**

问题一：图形界面如果纯用OpenGL完成将会过于复杂

解决方法：使用Swing创建，并通过自定义界面样式保持界面的美观

**完整过程**

1. 小组讨论主题
2. 确定项目的基本开发方式，选择图形界面的实现方式
3. （项目完成后补充）

**设计思想**

以面向对象的设计思想开发。同时运用Symbol Table、Binary Search Tree等数据结构和算法优化性能，让程序拥有迅速完成用户操作的能力，减少用户等待时间。并添加从实际观察中发现能够解决实际问题的功能。