# 数据库课程设计实验心得

第六组16340198 孙肖冉

1. 数据结构的设计
2. 一开始构想时完全不理解如何实现磁盘中的页的形式，在TA的提醒以及其他组同学的帮助下，我们设计了两个类Vector （用来存储向量的信息:id,空间位置），Page（包含页码，vector数组（slot），以及向量的信息（以数组的形式进行存储））。一开始设计类的这种结构时，也考虑到了这种结构可以在完成第二部分时增添所需的函数。
3. 根据要求我们通过计算可以得到MNIST每页可以存储20个向量及其信息，glove每页可以存储54个向量及信息；因为两组数据的读写方式基本相同，所以在实现时，虽然用不同的代码文件，但是只需要改变一些宏定义值的量即可。
4. 在查询最近值时我们设计了一个Node（存储向量的id，以及与测试向量的距离distance）
5. 二进制文件读写

为了实现在磁盘中的存储，所以文件在读取写入时要以二进制的形式。

在进行实验之前，并没有接触过二进制文件读写的有关知识。所以花费了一点时间去查找相关资料

在本次实验中主要运用了：fopen(),fscanf(),fread(),fwrite(),fseek(),fclose()，rewind()函数

在利用fseek()函数进行文件指针移动之后，然后利用fread()对二进制的文件进行读取数据的操作时，得到的仍然是文件开头的数值。后来找到了解决方法，但是仍是不能理解。先写入心得中以防之后忘记。



每个函数成功时的返回值不尽相同，例如fseek()的返回值是0（被这个坑了好久）。

函数的实现在网上都有很多资料，在此就不赘述了。

1. 性能的优化

在这次实现返回最近点的时，程序所运行的时间花费过长，跑1000个向量的时间太长。猜想是因为在从磁盘中读取数据时，读取一个向量信息之后及进行处理，然后试验了另一种方式，每次读取一页数据然后在进行数据处理，发现所用时间更长了。最后还是保留了第一种方法。从其他组的同学那得到了另一种方式的灵感，但是因为时间原因，无法完全更新实现，所以在下一次实验中尝试进行优化。

本次实验的难点其实在于对于题目的理解，如何实现页的结构。至于二进制文件的读写，反而是因为是具体的函数，能够搜寻到许多资料实验并不难。但是根据读取的方式不同，程序运行时间仍然会存在很大差别。