# 数据结构研讨课

1651459 宋宇凡

**研讨课内容的总结**

我们组对三个问题都进行了讨论，都讨论出了结果，其中我们最主要讨论的第三个问题。

**第三题 有一千万条短信，有重复，以文本文件（ASCII）形式保存，一行一条，请找出重复出现最多的前十条。**

**算法思考思路：**

1．首先，当第一眼看到这个问题的时候，我们组第一反应是遍历查找匹配，但是该算法的时间复杂度是O(N)，所以我们组很快否决了这个算法。

2．之后，我们想采取哈希表的算法，对这个数据进行哈希映射，然后建立相对应的哈希树，之后进行操作，但是我们考虑到一千万条信息，而且都是用ASCII码表示，而且这个题目的要求是要对数据进行重复性的统计，那么，我们组觉得使用哈希的话，对于哈希结果冲突以及对重复性结果的统计是一处理起来比较麻烦的情况，所以我们也放弃了这个思路。

3．最后，我们组提出一个观点，使用MD5码。

**算法详细分析解释：**

首先，我们先建立一个树，初始化的时候只有一个根节点，然后对所用的数据的每一条，都进行如下的操作。

先对一条信息进行一个映射的处理，之后得到其对应的MD5码。

对这个MD5码进行两个操作，首先在树中进行查找匹配操作，对这个128位的数值的每一位分别依次进行一下操作，从树的根节点开始，如果根节点的子节点中有这位数值，那么进入到这个节点中进行循环操作，当循环结束，在叶节点中，对叶节点的计数属性+1，表示发现了一条重复的信息。如果没有这个节点，那么开辟一个新的节点，数值等于这个未找到匹配项的数值，然后将后序所有的数值依次申请节点插入成为子节点，当这128位数值的每一位都遍历完，即到达叶节点，同时设置这个叶节点拥有一个记录次数的属性，用于统计重复次数。

当对一个MD5码操作完成之后，我们将会得到这个MD5码目前的重复次数，然后我们另建立一个数组在这个数组中，我们将动态的不断进行比较，从而记录目前前十个最大的重复次数，同时也记录这些前十个对应的文本（虽然MD5码目前可以破解，但是破解将花费很长的时间，所以在这里对去进行记录操作，从而当遍历完成之后，我们也将得到最大的前十个文本）。（此处由于所给文本读取的时候有一定的顺序，也可以考虑下面一个方法来取代排序，在每一个叶节点的位置上加入一个位置属性，用于记录此MD5码所对应的文本中第几条文本信息，最后当遍历完所有信息之后，对生成的树进行一个遍历，访问叶节点的次数属性，和索引位置属性，从而得到最大的前十个，之后也可以直接对应到文本信息中，也可以得到最大的前十个文本信息，但是考虑到如果对每一个结点都添加一个位置属性，那么由于树的节点很多，128\*1千万个int数据将会额外多占用大量的空间，所以此处不采取这种方式）。

**采用算法与其他算法比较：**

1. 优势：

由于Md5码映射出来的结果是一个固定的位数，所以所建树的深度是固定的，与N无关，这意味着在查找操作时，也是O(1)的复杂度。这种优势在对于大量的单条数量量很大的文本处理时，会有很大的优势。

2．缺点：

MD5码目前可能出现的碰撞问题，但是由于MD5是一个128位的散列值，所以即使在一千万的数据样本下，这个问题出现重复的概率仍十分的低，可以忽略不计。

如果每一条短信的长度较短，只有十个左右的字符，那么使用MD5码的效率会低于不对其进行处理的效率，在一千万的样本下，时间复杂度和空间复杂度虽然都一样，但是由于MD5的位数固定，所以会在时间和空间上多花一些时间，之后也许可以对算法进行优化，根据数据情况进行分类，根据文本的长度，从而对其进行不同的处理。

3.与哈希树进行比较：使用MD5和选用更大的质数相结合的办法。这样就可以使得通过层次比较少的哈希树来获得对关键字区间的完整覆盖。这样就减少了比较操作的次数，并提高整体的工作效率。所以其实采取MD5码相当于是对哈希树的一个优化。

4.相比于字典树来说，我觉得，MD5码在对于文本信息比较长时有更大的优势，鉴于题中给出的是ASCII码，虽然说文本信息只有一行，但是由于考虑到文本文件形式的一行可以有很长，所以如果采取字典树的形式，那么这颗树会变得很深，但是如果使用MD5码的形式，由于MD5码位数固定，这棵树并不会很深，我觉得这是一个优势，但是如果文本信息较短，那么采取字典树的形式将会有更好的效果。

**对教学和研讨课的建议**

我觉得，这一次的研讨课比上一次的研讨课的题目出的更能激起同学们讨论的欲望，上一次的研讨课的题目更多的还是类似于书本上的算法和题目，即使是医院排队管理的问题，也更多的难以在算法上进行很好的处理，但是这一次的研讨课的题目不一样，第一题是一个AVL树的数据结构，第二题是Windows下的一个算法现实，第三题则涉及到了哈希表，哈希树，字典树的应用，我觉得这些问题更能激起同学们的讨论兴趣，开阔大家的视野，同时我也希望对于这一类型的题，老师可以在研讨课上多出一些问题，这样大家讨论起来更好（不然容易讨论每多长时间就讨论完了），而且不会出现讲题的时候，大家都讲一个题的现象。而且如果题多了，涉及到的算法范围和种类也必然更加的多，我们在和同学们交流研讨过程中也就能有更多的机会来学习自己不会的，但是其他同学知道的算法，这样对大家的水平的提升也有很大的帮助。