**算法分析与设计实验报告**

**第 一 次实验**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 姓名 | 李平凡 | 学号 | 201907040102 | | 班级 | 计科1905 |
| 时间 | 3.21 | 地点 | 软件大楼 | | | |
| 实验名称 | 字典序问题 | | | | | |
| 实验目的 | 通过搜索资料，要求掌握一些基本问题和典型例题的处理方式 | | | | | |
| 实验原理 | 求一个字符串的字典序，按照题意可知一个n位的字符串一定是在比它小的字符串位数之前，也就是说前面的位数少的字符串以一种有序的方式进行了排列那么就用到组合数，其次就是n位字符串自身的排列，n位字符串自身的排列需要从高位开始按顺序查找相应的字母来判断后面的n-1位由多少个字母进行组合。 | | | | | |
| 实验步骤 | ① 先知道字符串长度，然后在n位长度前有1到n-1位的字符串长度的有序排序。  ② 将该n位字符串第一位开始寻找单个字母序比它小的字母的n位字符串，可知其后面的n-1位必然是这个字母以后的所有字母的有序排列。用这种方式去找次高位。  ③ 通过循环遍历所有位得到最后的sum结果但需要注意的是当字符串只有一位的时候，后面是0位并且比1位小的字符串也是0，所以需要分离开着一种情况。 | | | | | |
| 关键代码 | int pailie(int m,int n) //算排列数  {  if(n==0)  {  return 0;  }  int km=1,kn=1;  for(int i=1;i<=n;i++)  {  kn=kn\*i;  }  for(int i=m-n+1;i<=m;i++)  {  km=km\*i;  }  return km/kn;  }  for(int i=1;i<str.size();i++)  {  sum=sum+pailie(26,i);  }  n位字符串前面有多少种情况  for(int i=0;i<str.size();i++)  {  a[i]=str[i]-'a';  }  int temp=1; //temp存储前面有多少个字母不能用  for(int i=1;i<=str.size();i++)  {    for(int j=temp;j<=a[i-1];j++)  {  sum=sum+pailie(26-j,str.size()-i);  }  temp=2+a[i-1];  }  用a数组存储每一位和’a’相比有多少个空位，temp存储多少个字母不能用，通过排列后面的n-1位字母不断相加得到最终结果。 | | | | | |
| 测试结果 |  | | | | | |
| 实验心得 | 此问题时间开销较少，主要是因为采用好的算法，一般算法思想是通过将所有排列排出来，然后找到其中的字符串从而确定其字典序，但这种方法不仅耗时从6位字典序已经看出来达到十万级别了，所以时间开销很大而且还很难操作将每一个字符串列出来。由此看出算法的重要性，因为该算法与字符串的位数紧密相连，通过位数进行数学运算得到结果，由于位数为n且其中用到了组合数的函数，时间复杂度为O(n^2)但由于是字母组合数保证了n必然小于26所以时间开销并不是很高。 | | | | | |
| 实验得分 |  | 助教签名 | |  | | |

**附录：完整代码**