**《数据结构》实验报告（第2章）**

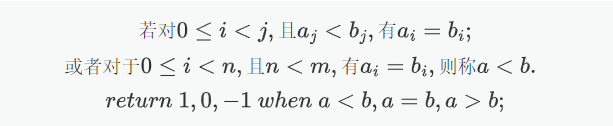
**学号：\_09020328\_\_\_\_ 姓名 ：\_ 王亮\_\_\_\_**

**实验题号： P93T2 P107T4 P118T1\_ 实验日期：\_2021.08.18\_\_\_**

**实验一**

**1．问题描述：**

比较a 和 b两有序表大小，遵循如下图规则：



（注：latex放word比较麻烦，效果不大好，就放截图了）

**2．算法思想：**

从索引0开始逐位比较大小，若有一位a > b直接返回 1，反之-1，若每一位均相同，则比较位数大小，若a位数大于b，返回1反之-1，若此时位数相等，返回0

**3．功能函数：**

Lab2():封装比较随机长度相同数据下a，b大小

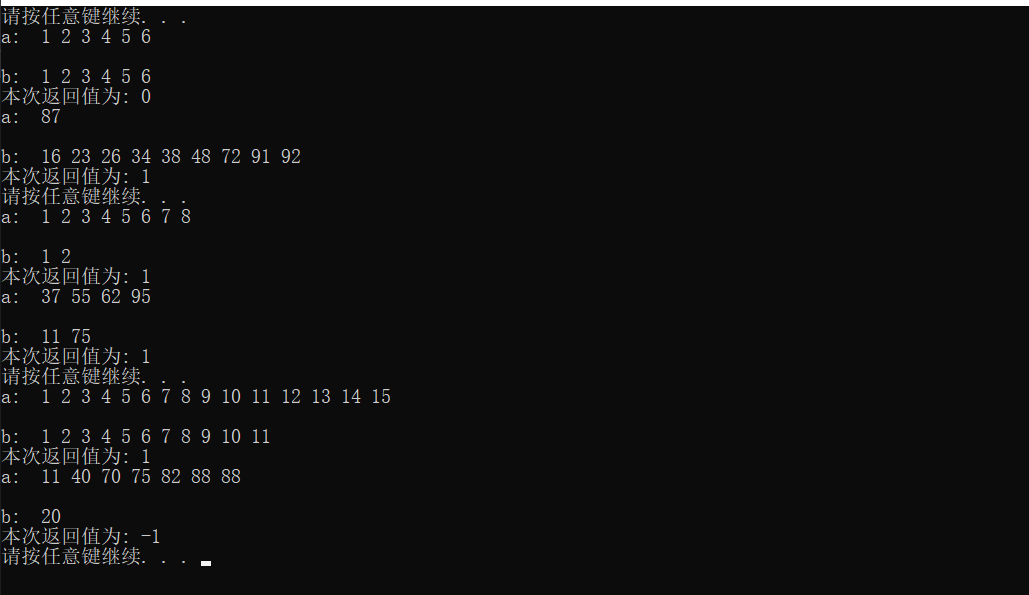
Lab2(int):封装比较随机长度，随机数据下a，b大小

Compare:比较函数，按上方算法思想描述的规则比较a，b并返回相应结果

**4．测试数据：**

由rand()函数生成，更体现一般性。

**5．测试情况：**。



**6．实验总结：**

计a长度：n，b长度：m

时间复杂度均为O（min{m,n}）

空间复杂度O（1）{无额外空间}

**7. 源代码：**

实验源程序清单

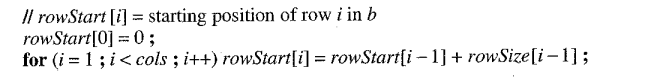
(1) As-LabP93T2.cpp

**实验二**

**1．问题描述：**

重写FastTranspose函数，使其利用一个数组存储RowSize和RowStart。

**2．算法思想：**



由书本提供的上述推导式可知：

从1开始：rowStart[i] = rowStart[i-1] + rowSize[i-1],且rowStart[0]=0,rowSize[0]已知

故可提供两临时变量进行如下操作：

t = rowStart[0]; //获取rowStart[0]

for(i=0;i<cols;i++):

s = rowSize[i]; //获取rowSize

rowSize[i] = t; // 用rowStart和rowSize计算结果更新rowStart

t = t + s; //计算下一次的rowStart结果

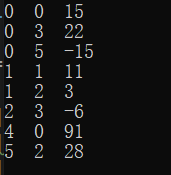
即可得到最后所需rowStart(存放在rowSize数组中)

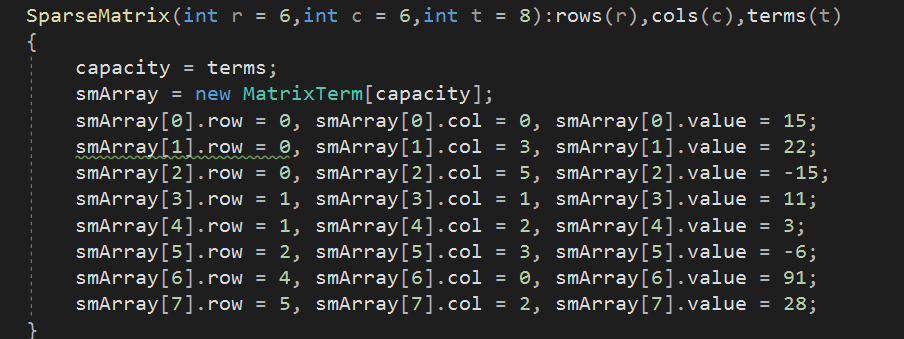
**3．功能函数：**

Lab4():封装输出与FastTranspose（）调用

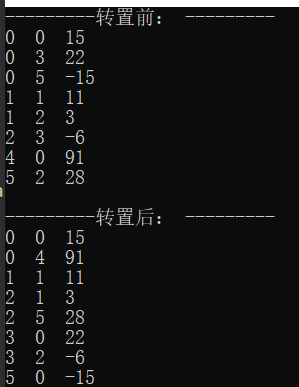
FastTranspose（）：使用上述思想进行快速转置

**4．测试数据：**





**5．测试情况：**



**6．实验总结：**

时间复杂度：O（terms+cols）

空间复杂度：O（n）{n为矩阵cols（列数）}

**7. 源代码：**

实验源程序清单

(1) As-LabP107T4.cpp

**实验三**

**1．问题描述：**

编写一个确定字符串中不同字符各自出现频率的函数，并用适当的数据进行测试。**2．算法思想：**

建立一个与ASCII码编号一一对应的128大小哈希表，

对应字符编号位置记录其出现频数

然后遍历字符串，让遍历到的字母处的频数自增；

**3．功能函数：**

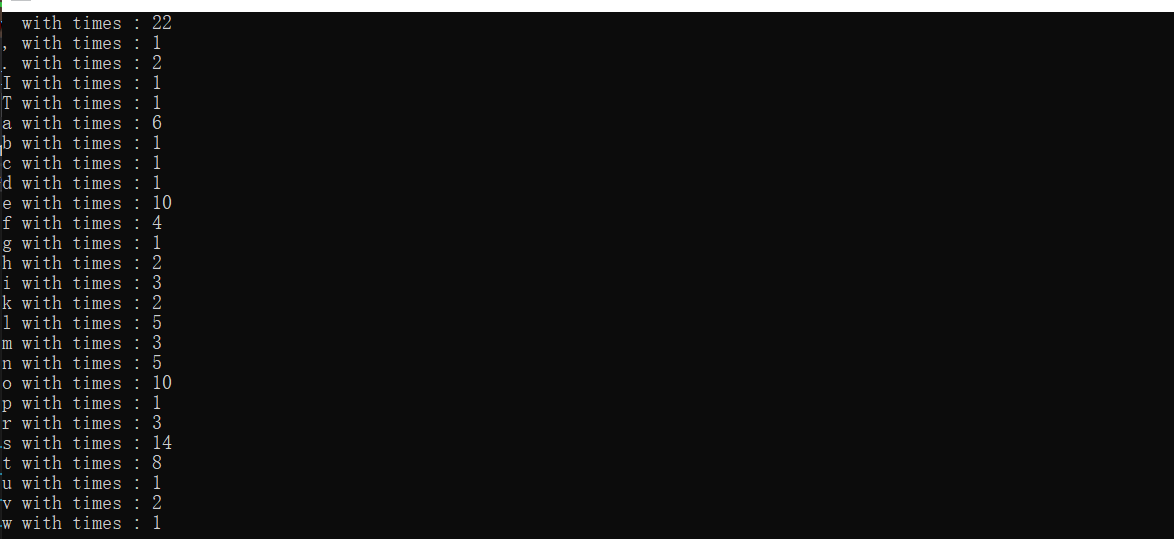
Process（）：包含遍历与输出

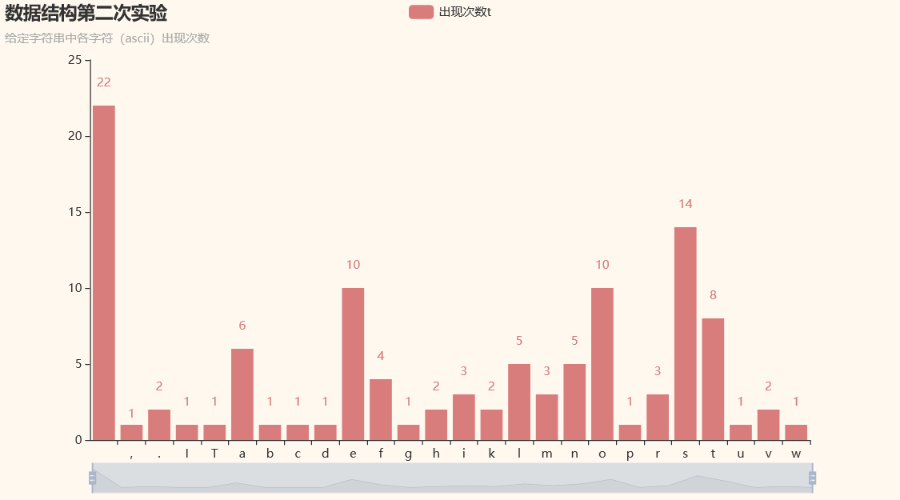
**4．测试数据：**

"The world puts off its mask of vastness to its lover. It becomes small as one song, as one kiss of the eternal."

**5．测试情况：**

可采用Regex101进行验证





**6．实验总结：**

记字符串长度为n：

时间复杂度： O（n）

空间复杂度：O（1）{ASCII码从0-127，使用大小为128的哈希表，常量级空间}

**7. 源代码：**

实验源程序清单

(1) As-LabP118T1.cpp