

**代码实现如下：**

int **frequency**(char \*sStr,char arrChar[],int arrTimes[])

{

*//arrChar 记录字符串中所有不同的字符*

*//arrtimes记录出现次数*

int count = 0; *//记录字符的个数*

int len = **strlen**(sStr);

if(len == 0) return count;*//字符串为空*

for(int i = 0; i < len; i++)

arrTimes[i] = 0;

for(int i = 0; i < len; i++)

{

int flag = 0;

for(int j = 0; j < count; j++)

{

if(arrChar[j]==sStr[i])

{

arrTimes[j]++;

flag++;

break;

}

}

if(!flag)

{

arrChar[count] = sStr[i];

arrTimes[count++]++;

}

}

return count;

}

**主函数中添加测试：**

int **main**()

{

char\* cStr = new char[100];

char arrChar[100];

int arrTimes[100];

cout **<<** "输入cStr:" ;

cin **>>** cStr;

int count = **frequency**(cStr,arrChar,arrTimes);

cout **<<** "频度：" **<<** count **<<** **endl**;

cout **<<** "字符：个数，如下：" **<<** **endl**;

for(int i = 0; i < count; i++)

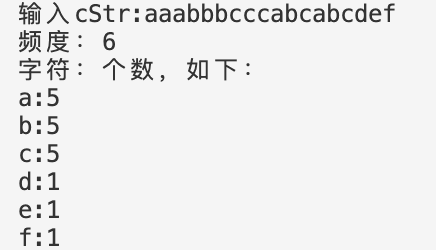
cout **<<** arrChar[i] **<<** ":"**<<**arrTimes[i]**<<endl**;

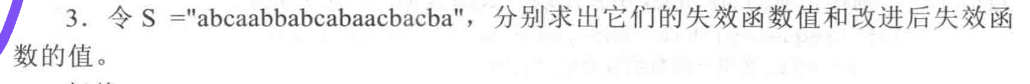
delete[] cStr; *// 释放内存*

return 0;

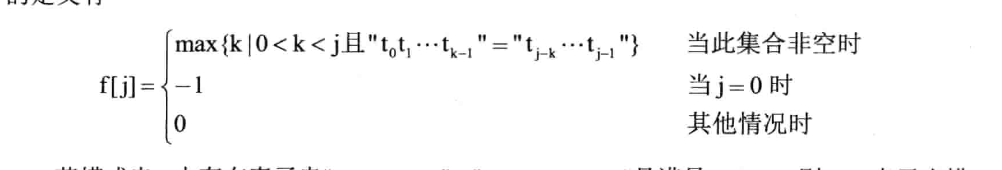
}

**测试结果如下：**



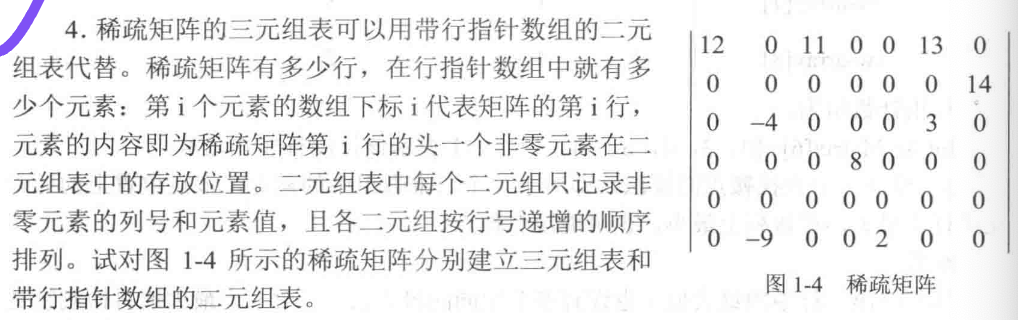


**根据书上的公式：**



|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| j | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| s [ j ] | a | b | c | a | a | b | b | a | b | c |
| f [ ] | -1 | 0 | 0 | 0 | 1 | 1 | 2 | 0 | 1 | 2 |
| 改进 | -1 | 0 | 0 | -1 | 0 | 0 | 2 | -1 | 0 | 0 |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| j | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 |
| s [ j ] | A | b | a | a | c | b | a | c | b | a |
| f [ ] | 3 | 4 | 2 | 1 | 1 | 0 | 0 | 1 | 0 | 0 |
| 改进 | -1 | 4 | 2 | 1 | 1 | 0 | -1 | 1 | 0 | -1 |



**三元组表如下：**

|  |  |  |  |
| --- | --- | --- | --- |
| arr[ ] | row | column | val |
| 0 | 0 | 0 | 12 |
| 1 | 0 | 2 | 11 |
| 2 | 0 | 5 | 13 |
| 3 | 1 | 6 | 14 |
| 4 | 2 | 1 | -4 |
| 5 | 2 | 5 | 3 |
| 6 | 3 | 3 | 8 |
| 7 | 5 | 1 | -9 |
| 8 | 5 | 4 | 2 |

**二元组表如下：**

|  |  |  |
| --- | --- | --- |
| arr[ ] | column | val |
| 0 | 0 | 12 |
| 1 | 2 | 11 |
| 2 | 5 | 13 |
| 3 | 6 | 14 |
| 4 | 1 | -4 |
| 5 | 5 | 3 |
| 6 | 3 | 8 |
| 7 | 1 | -9 |
| 8 | 4 | 2 |

**行指针数组**

{0，3，4，6，-1，7}

