

温州大学计算机与人工智能学院

程序设计基础(2023-2024-1) 课程作业

实验名称	单词统计(I)				
班 级	23大数据1	姓 名	徐王晶	学 号	23211870102
实验地点	南5B105	实验时间	2023-10-24,16:50:43	指导老师	虞铭财

一、问题编号：

0448

地址: <http://10.132.254.54/problem/448/>

二、问题描述：

给定一段文章，每行不超过1000个字符，统计文章中每个单词出现的频率，忽略大小写差异。为了简化问题，文章中只有英文字母和空白字符，没有标点符号和特殊符号。

三、输入说明：

第一行为一个整数T，表示数据的组数。 对于每组数据以一个独占一行的“#”结束。

四、输出说明：

对于每组测试数据： 第一行为一个整数n表示文章中出现的单词的总数； 接下来n行按字母序从小到大输出文章中出现的单词 (全部以大写字母表示) 和出现的次数。

五、输入样例：

```
2
Hello World
#
Abc abc cde aa bb cc aas test
#
```

六、输出样例：

```
2
HELLO 1
WORLD 1
7
AA 1
AAS 1
ABC 2
BB 1
CC 1
CDE 1
TEST 1
```

七、解答内容：

所用语言：

源代码：

```
01. #include <stdio.h>
02. #include <string.h>
03.
04. void upper(char * word);
05.
06. int main(void)
07. {
08.     int n;
09.     scanf("%d", &n);
10.
11.     while (n--)
12.     {
13.         char word_count[100][100];
14.
15.         int count[100] = {0};
16.         int p = 0;
17.         char word[100];
18.         scanf("%s", word);
19.         upper(word);
20.         for(int i=0; i<100; i++)
21.             if(word_count[i][0] == 0)
22.                 word_count[i][0] = word[0];
23.         for(int j=1; j<100; j++)
24.             if(word_count[j][0] == word[0])
25.                 count[j]++;
26.         else
27.                 count[p]++;
28.         p++;
29.     }
30.     for(int i=0; i<p; i++)
31.         printf("%s %d\n", word_count[i][0], count[i]);
32. }
```

```

18.     upper(word);
19.     while (word[0] != '#')
20.     {
21.         if (p == 0)
22.         {
23.             strcpy(word_count[p], word);
24.             count[p] = 1;
25.             p++;
26.         }
27.         else
28.         {
29.             for (int i = 0; i < p; i++)
30.             {
31.                 if (strcmp(word, word_count[i]) == 0)
32.                 {
33.                     count[i] += 1;
34.                     break;
35.                 }
36.                 else if (i == p - 1)
37.                 {
38.                     strcpy(word_count[p], word);
39.                     count[p] = 1;
40.                     p++;
41.                     break;
42.                 }
43.                 else if (strcmp(word, word_count[i]) < 0)
44.                 {
45.                     for (int j = p; j > i; j--)
46.                     {
47.                         strcpy(word_count[j], word_count[j - 1]);
48.                         count[j] = count[j - 1];
49.                     }
50.                     strcpy(word_count[i], word);
51.                     count[i] = 1;
52.                     p++;
53.                     break;
54.                 }
55.                 else if (
56.                     strcmp(word, word_count[i]) > 0
57.                     && strcmp(word, word_count[i + 1]) < 0
58.                 )
59.                 {
60.                     for (int j = p; j > i + 1; j--)
61.                     {
62.                         strcpy(word_count[j], word_count[j - 1]);
63.                         count[j] = count[j - 1];
64.                     }
65.                     strcpy(word_count[i + 1], word);
66.                     count[i + 1] = 1;
67.                     p++;
68.                     break;
69.                 }
70.             }
71.         }
72.         scanf("%s", word);
73.         upper(word);
74.     }
75.
76.     printf("%d\n", p);
77.     for (int i = 0; i < p; i++)
78.     {
79.         printf("%s %d\n", word_count[i], count[i]);
80.     }
81. }
82.
83.
84. void upper(char * word)
85. {
86.     for (int i = 0; i < strlen(word); i++)
87.     {
88.         if (word[i] >= 'a' && word[i] <= 'z')
89.         {
90.             word[i] -= 32;
91.         }
92.     }
93. }

```

八、判题结果

AC - 答案正确

判题结果补充说明：

test id: 839,result:AC, usedtime:0MS, usedmem:1036KB,score:50 test id: 897,result:AC, usedtime:0MS, usedmem:1036KB,score:50