## 实验八　字符数组程序设计

### 一、实验学时

3学时

### 二、实验目的

（一）掌握字符数组的定义、初始化和应用；

（二）掌握字符串处理函数的使用。

### 三、预习要求

重点预习的内容：C语言中字符串的存储表示；字符数组输入输出的方法；常用的字符串处理函数的使用。

### 四、实验内容

（一）输入下面的程序并运行，观察程序运行的结果，并分析原因（注意程序第2行中有些单引号之间是空格）。

/\* c8-1.c 字符数组的输出\*/

#include "stdio.h"

int main( )

{

char a[10]={ ’I’, ’ ’, ’a’, ’m’, ’ ’, ’a’, ’ ’, ’b’, ’o’, ’y’};

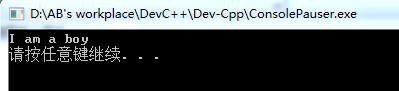
printf("%s\n",a);

return 0;

}

将字符数组a的大小改为11，再运行程序，并将结果与修改前的结果进行比较，分析原因。

**10:**

****

**11:**

**原因分析：字符串长度为10，但需要11个元素，最后一个用于存放字符串结束符’\0’，printf函数输出字符串时检测结束符’\0’并输出其之前的内容。定义字符数组长度为10时，无法检测到结束符，继续输出未知内容。**

（二）按照要求编写程序：有一行文字，不超过80个字符，分别统计出其中英文大写字母、小写字母、数字、空格、及其它字符的个数。

*编程提示：*

1．定义一个一维字符数组。

2．定义5个整型变量分别统计大写字母、小写字母、数字、空格和其它字符的个数（即作为5个计数器使用），并为这5个变量赋初值。

3．用scanf(输入的空格变成\0）函数或gets函数为字符数组赋一个字符串。

4．在循环中对字符数组的每个元素进行判断，相应的计数器加1。注意循环控制的条件和进行判断的条件怎样设置。

5．循环结束后输出各计数器的值。

思考：如果是对一篇英文文章进行统计,又该怎么编程呢？文章的行数和每行字数可以自己来设。提示：对文章的内容要用二维字符数组来存储。

#include<stdio.h>

#define LINE 5 //最大行数

#define COLUMN 80 //每行最多字符

//Think1

int main()

{

char a[LINE][COLUMN];

int uplt, lwlt, num, blk, other, i, j; //定义4个计数器 以及 循环控制变量

uplt = lwlt = num = blk = other = 0;

for(i = 0; i < LINE; i++) //输入文本

gets(a[i]);

for(i = 0; i < LINE; i++){ //嵌套循环遍历二维数组

for (j = 0; a[i][j] != '\0'; j++){

if (a[i][j] > 'A' && a[i][j] < 'Z') uplt++;//计数

else if (a[i][j] > 'a' && a[i][j] < 'z') lwlt++;

else if (a[i][j] > '0' && a[i][j] < '9') num++;

else if (a[i][j] == ' ') blk++;

else other++;

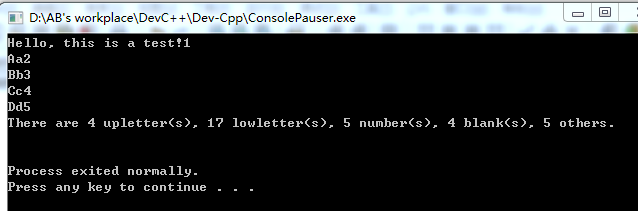
}

}

printf("There are %d upletter(s), %d lowletter(s), %d number(s), %d blank(s), %d others.\n",uplt, lwlt, num, blk, other);

return 0;

}



/\* c8-2.c 统计字符个数 \*/

#include<stdio.h>

int main()

{

char a[20];

int uplt, lwlt, num, blk, other,i;

uplt = lwlt = num = blk = other = 0;

gets(a);

//printf("\*%s\n",a); //调试用

for(i = 0; a[i]!='\0'; i++){

if (a[i] > 'A' && a[i] < 'Z') uplt++;

else if (a[i] > 'a' && a[i] < 'z') lwlt++;

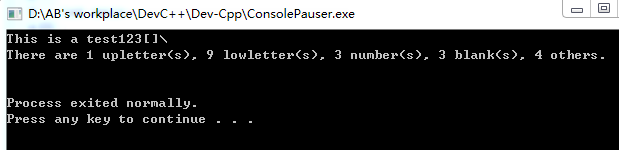
else if (a[i] > '0' && a[i] < '9') num++;

else if (a[i] == ' ') blk++;

else other++;

}

printf("There are %d upletter(s), %d lowletter(s), %d number(s), %d blank(s), %d others.\n",uplt, lwlt, num, blk, other);

 return 0;

}

（三）下面程序的功能是实现将两个字符串连接起来并输出结果，注意不使用strcat函数。请填空并运行程序。

*编程提示：*

1．定义两个一维字符型数组str1、str2和两个循环变量。

2．为两个字符数组输入两个字符串（可使用scanf函数或gets函数整体赋值，要注意scanf和gets函数的区别，在对字符串赋值时，scanf函数不能出现空格）。

3．确定字符数组str1结束的位置。

4．再将字符数组str2中的内容连接到字符数组str1的后面。

5．为字符数组str1赋字符串结束的标志’\0’。

6．输出连接后的字符数组str1。

/\* c8-3.c 字符串连接\*/

#include "stdio.h"

int main( )

{

char str1[100],str2[100];

int i=0,j=0;

printf("please input the string1:");

gets(str1);

printf("please input the string2:");

gets(str2);

for(i=0; str1[i]!='\0'; i++) ; /\*注意,此处空语句不可少\*/

for(j=0;str2[j]!='\0';j++){

str1[i]=str2[j];

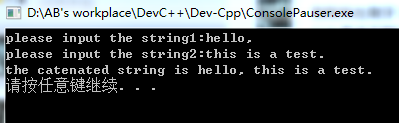
i++;

}

**str1[i+1] = '\0';**  /\*给出新的字符串的结束符\*/

printf("the catenated string is %s",str1);

return 0;

}

（四）下面程序的功能是用strcat函数实现将字符串2连接到字符串1的后面并输出，请补充完整。

/\* c8-4.c 字符串连接\*/

int main( )

{

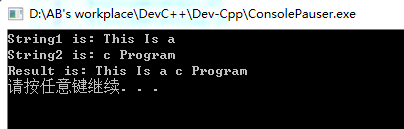
char str1[80]="This Is a ",str2[80]="c Program";

printf("String1 is: %s\n",str1);

printf("String2 is: %s\n",str2);

**strcat(str1,str2);** /\*使用strcat函数实现，注意其格式\*/

printf("Result is: %s\n",str1);

return 0;

}

（五）下面程序的功能是实现将一个字符串中的所有大写字母转换为小写字母并输出，请补充完整。

例如：当字符串为"This Is a c Program"

输出："this is a c program"

/\* c8-5.c 字符串中的大写字母转为小写字母\*/

#include "stdio.h"

#include<stdlib.h>

int main( )

{

char str[80]="This Is a c Program";

int i;

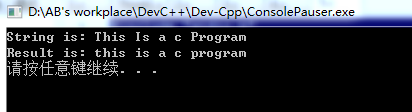
printf("String is: %s\n", str);

for (i = 0; str[i]!='\0'; i++)

if (str[i] >= 'A' && str[i] <= 'Z')

str[i] += 32; /\*将大写字母转换成小写字母\*/

printf("Result is: %s\n",str);



system("pause");

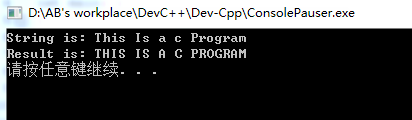
return 0;

}

思考：如果将字符串中的所有小写字母转换为大写字母，又将如何修改程序？

if (str[i] >= 'a' && str[i] <= 'z')

str[i] -= 32; /\*将小写字母转换成大写字母\*/



（六）编写程序实现在一个字符串中查找指定的字符，并输出指定的字符在字符串中出现的次数及位置，如果该字符串中不包含指定的字符，请输出提示信息。

*编程提示：*

1．定义两个一维数组，a字符数组用来存放字符串，b整数数组用来存放指定的字符在字符串中出现的位置(即对应的下标)。

2．定义i,j,m三个循环控制变量和一个标志变量flag，并初始化flag的值为0。

3．用scanf或者gets函数为字符数组赋一个字符串。

4．在循环中对字符数组的每个元素和指定字符ch进行匹配判断，如果相同，就把其下标依次存放在数组b中，并置flag的值为1。

5．循环退出后判断标志变量flag的值，如果仍为0，说明字符串中没出现指定的字符，否则，就输出该字符在字符串中出现的次数和位置。

#include<stdio.h>

#include<stdlib.h>

int main()

{

char a[50]; //定义a存放字符串

int b[50]; //存放指定字符位置

int flag=0, i, j=0,k, ch; //定义标志变量、循环变量

printf("\n输入文本：");

gets(a);

printf("输入要查找的字符： ");

ch = getchar();

//puts(a); //调试用

for(i = 0; a[i] != '\0'; i++){

if (a[i] == ch){ //找到指定字符

b[j] = i; //将下标存放在指定数组

flag = 1; //标志变量置1表示找到指定字符

j++;

}

}

k = j; //保存b的有效位

if (flag == 0) printf("没有找到该字符\n");

else{

printf("指定字符%2c出现了%3d次\n位置是： ",ch,j);

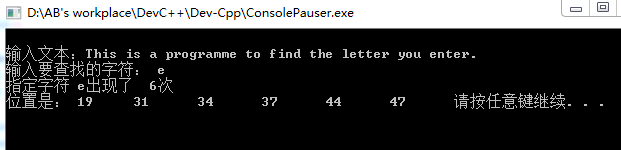
for(j = 0; j < k; j++)

printf("%d\t",b[j]+1);

}

system("pause");

return 0;

}

### 五、实验注意事项

（一）注意C语言中字符串是作为一维数组存放在内存中的，并且系统对字符串常量自动加上一个‘\0’作为结束符，所以在定义一个字符数组并初始化时要注意数组的长度。

（二）注意用scanf函数对字符数组整体赋值的形式。