# 实验报告7

实验七名称：字符串 练习

实验目的：1、熟悉字符数组的概念，掌握字符数组处理字符串的基本应用。2、掌握字符数组名做函数参数的设计使用 。

填写日期： 2022年11月24日

本次实验成绩：

1. **实验任务：**

1、编写一个函数实现字符串长度的统计。

2、编写程序，要求从键盘输入一任意长度字符串，然后将该字符串中的每个字符反向输出。（提示：先求出字符串长度，然后从最后一个字符开始往前逐个输出字符）

3、输入两个字符串（串长度需大于5），将第一个字符串的前5个字符与第二个字符串的前5个字符组成一个新的字符串，输出这个新的字符串。

4、编程发现以下字符串中出现的统计规律，例如不同三字符子串“agg”、“ggc”等出现的次数，统计字符串为：“aggcacggaaaaacgggaataacggaggaggacttggcacggcattacacggaggacgaggtaaaggaggcttgtctacggccggaagtgaagggggatatgaccgcttgg”。

1. **实验方法及知识点应用：**

1、字符数组的定义与赋值（初始化、循环键盘输入），下标法访问字符数组元素的方法。

2、字符数组的典型应用（复制、连接、搜索统计某种元素的个数）。

3、字符数组存储字符串的特点：‘\0’字符的使用。

4、字符数组名作为函数参数的特点与应用方法。

1. **实验程序代码及结果记录：（以表述清楚为目的）**

任务1：

源代码：

#include<stdio.h>

char str[100000];

int getCharNum(char[]);

int main()

{

printf("请输入字符串：\n");

gets(str);

int num = getCharNum(str);

printf("字符串长度为：%d", num);

return 0;

}

int getCharNum(char a[])

{

int i, sum = 0;

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

{

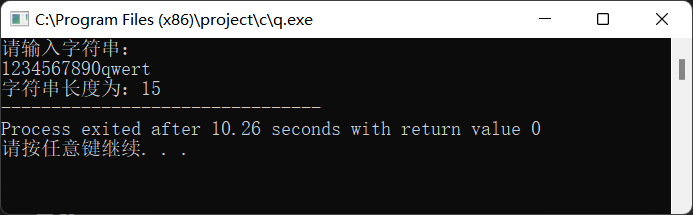
sum++;

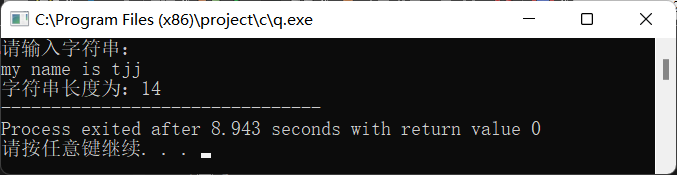
}

return sum;

}

测试：





任务2：

源代码：

#include<stdio.h>

#include<string.h>

char str[1000000];

int main()

{

printf("请输入字符串：\n");

gets(str);

int i;

printf("逆序输出字符串为:\n");

for(i=strlen(str)-1;i>=0;i--)

{

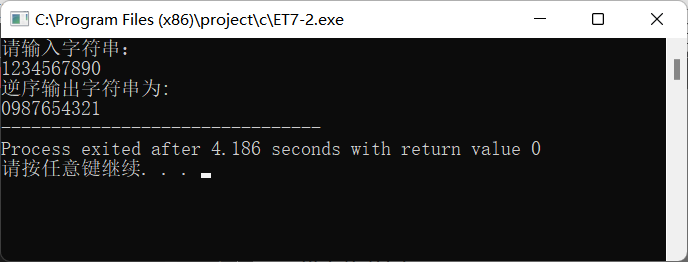
printf("%c",str[i]);

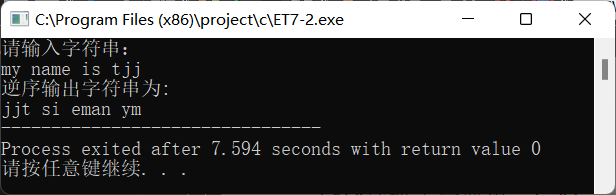
}

return 0;

}

测试：





任务 3：

源代码：

#include<stdio.h>

#include<string.h>

char str1[1000000];

char str2[1000000];

int main()

{

printf("请输入字符串1：\n");

gets(str1);

printf("请输入字符串2：\n");

gets(str2);

char str3[100]={};

strncat(str3,str1,5);

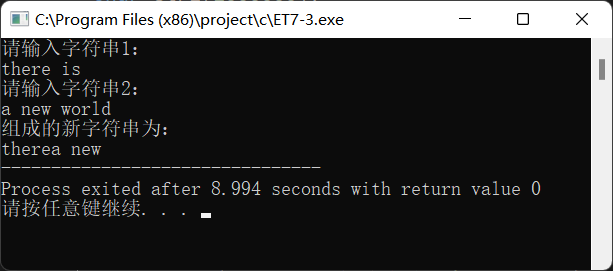
strncat(str3,str2,5);

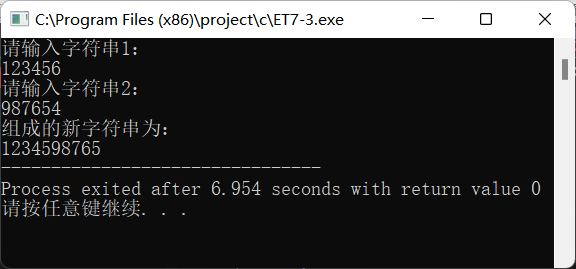
printf("组成的新字符串为：\n%s",str3);

return 0;

}

测试：





任务4：

源代码：

#include<stdio.h>

#include<string.h>

int main()

{

int sumagg=0,sumgga=0,sumggc=0;

int sumcgg=0,sumaaa=0,sumggg=0,sumata=0;

char \*str="aggcacggaaaaacgggaataacggaggaggacttggcacgg"

"cattacacggaggacgaggtaaaggaggcttgtctac"

"ggccggaagtgaagggggatatgaccgcttgg";

int len=strlen(str);

int i;

for(i=0;i<len-2;i++)

{

if(str[i]=='a'){

if(str[i+1]=='g')

{

if(str[i+2]=='g')

{

sumagg++;

}

}

}

if(str[i]=='g'){

if(str[i+1]=='g')

{

if(str[i+2]=='c')

{

sumggc++;

}

}

}

if(str[i]=='g'){

if(str[i+1]=='g')

{

if(str[i+2]=='a')

{

sumgga++;

}

}

}

if(str[i]=='c'){

if(str[i+1]=='g')

{

if(str[i+2]=='g')

{

sumcgg++;

}

}

}

if(str[i]=='a'){

if(str[i+1]=='a')

{

if(str[i+2]=='a')

{

sumaaa++;

}

}

}

if(str[i]=='g'){

if(str[i+1]=='g')

{

if(str[i+2]=='g')

{

sumggg++;

}

}

}

if(str[i]=='a'){

if(str[i+1]=='t')

{

if(str[i+2]=='a')

{

sumata++;

}

}

}

}

printf("agg的个数\n%d\n",sumagg);

printf("ggc的个数\n%d\n",sumggc);

printf("gga的个数\n%d\n",sumgga);

printf("ata的个数\n%d\n",sumata);

printf("aaa的个数\n%d\n",sumaaa);

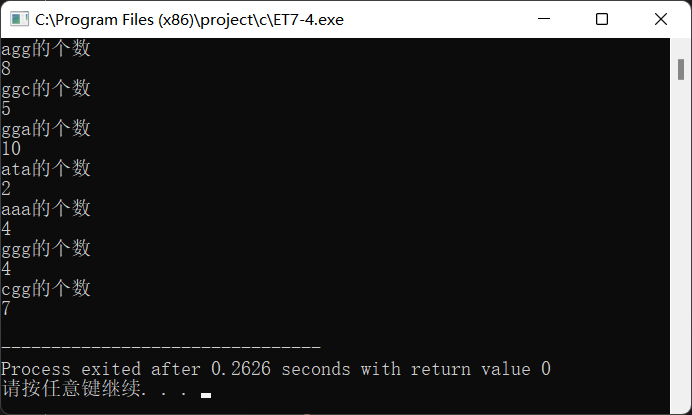
printf("ggg的个数\n%d\n",sumggg);

printf("cgg的个数\n%d\n",sumcgg);

return 0;

}

测试：



1. **实验小结：**

先总结自己应用的知识点，再总结自己对知识点的理解和分析，根据实验结果和记录，最后总结实验中常出现的错误和问题。

应用知识点：

用字符数组实现字符串，字符数组的定义与赋值，gets函数的使用，string.h库函数的使用来处理字符串（如strlen,strncat）,数组名作为函数参数，穷举法的应用，嵌套选择，字符数组的应用（复制、连接、搜索统计），‘\0’字符的使用，%s占位符的使用。

理解分析：

1. gets函数可以输入一串带空格的字符串，遇到换行才会结束，而scanf函数遇到空格和换行符都会停止输入，会导致字符串输入不全。
2. puts函数和printf函数通常能够互相替换，而gets和scanf却不能。
3. 字符串的结尾都会有一个‘\0’作为字符串结尾的标志，所以字符串实际长度要比表面长度多1，因此在设置字符数组长度时要注意多预留一位以防越界。
4. strlen函数能够计算字符串的长度，但‘\0’不会被记入，strcat函数可以将两个字符串进行连接并返回给第一个字符串strncat则是将第二个字符串的前n个字符与字符串1进行连接。
5. char\*和字符数组同样能够存放字符串，但char\*是只读的，无法对值进行修改，写成const char\*能更好表达它的意思。

注意：

1. 应用char\*构建字符串时一定要在定义时进行赋值，不能先定义后赋值，因为char\*本质上是一个指针，先定义的话不会指向任何内存空间，也就无法赋值，而定义时赋值则会存储在静态存储区，自动为其开辟内存空间。
2. 要输入带空格的字符串函数要用gets而不能用scanf.
3. 因为gets输入字符串是从字符数组的下标0开始存放，所以逆序输出时要strlen-1才不会越界。
4. 要用常量来设置数组的大小，不能用变量。