第9次作业答案

第7章

1. 编写个函数，由实参传递一个字符串，统计此字符串中字母、数字、空格和其他字符的个数，在主函数中输入字符串及输出上述统计的结果。

#include "stdio.h"

#define MAX 20

void calculateString(char \*pChar,int \*a);

void main()

{

char string[MAX]="abc 12,3";

int cacl[4];

int i;

for(i=0;i<4;i++)

{

cacl[i]=0;

}

printf("please input string!\n");

gets(string);

//输入字符串

calculateString(string,cacl);

printf("the string has %d char\n", cacl[0]);

printf("the string has %d number\n",cacl[1]);

printf("the string has %d blank\n", cacl[2]);

printf("the string has %d other\n", cacl[3]);

}

void calculateString(char \*pChar,int \*a)

{

while(\*pChar!='\0')

{

if ((\*pChar)>='a'&&(\*pChar)<='z') a[0]++;

else if ((\*pChar)>='A'&&(\*pChar)<='Z') a[0]++;

else if ((\*pChar)>='0'&&(\*pChar<='9')) a[1]++;

else if ((\*pChar)==' ') a[2]++;

else a[3]++;

pChar++;

}

}

2. 编写一个函数，在字符串中的指定位置插入一个子串，如在字符串"abcghi"中第三个 字符后插人子串"def"为"abcdefghi"，如插人位置不合法，原字符串不作任何处理;主函数完 成字符串、插入位置、子串的输入，调用所编函数得到插人后的字符串，并输出。

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

void insert(char \*parents,char \*son,int locate);

int main()

{

char str[100];

char substr[100];

int locate;

printf("请输入主字符串\n");

gets(str);

printf("请输入子字符串\n");

gets(substr);

printf("请输入位置\n");

scanf("%d",&locate);

insert(str,substr,locate);

return 0;

}

void insert(char \*str,char \*substr,int locate)

{

char temp[100];

int lenth;

lenth=strlen(str);

if(locate>lenth)

{

printf("得到的字符串为:\n%s\n",str);

}

else

{

strcpy(temp, str+locate);

\*(str+locate) = '\0'; //或者 strcpy(str + locate,"");

strcat(str,substr);

strcat(str,temp);

printf("得到的字符串为:\n%s\n",str);

}

}

1 实现函数char \*strstr(char \*s1,char \*s2); 并在main函数中调用该函数。比如在“Ever tried. Ever failed. No matter. Try again. Fail again. Fail better.”中查找“again”

其功能为：这个函数在s1中查找整个s2第一次出现的起始位置，并返回一个指向该位置的指针。如果s2并没有完整地出现在s1的任何地方，函数将返 回一个NULL指针。如果第二个参数是一个空字符串，函数就返回s1。

#include <stdio.h>

const char\* strstr(const char\* src, const char\* sub);

int main()

{

char str[80] = "Ever tried. Ever failed. No matter. Try again. Fail again. Fail better.";

char substr[20] = "again";

if (strstr(str, substr)==NULL)

printf("Not found.\n");

else

puts( strstr(str, substr) );

}

const char\* strstr(const char\* src, const char\* sub)

{

const char \*bp;

const char \*sp;

if(!src || !sub)

{

return src;

}

/\* 遍历src字符串 \*/

while(\*src)

{

/\* 用来遍历子串 \*/

bp = src;

sp = sub;

do

{

if(!\*sp) /\*到了sub的结束位置，返回src位置 \*/

return src;

}while(\*bp++ == \*sp ++);

src ++;

}

return NULL;

}