1. 将包含字符数字的字符串分开,使得分开后的字符串前一部分是数字后一部分是字母。例  
   如“h1ell2o3” ->”123hello”

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

int main()

{

char a[100];

char \*p,\*m;

char temp;

while(memset(a,0,sizeof(a)),gets(a)!=NULL)

{

p=a;m=a;

while(\*p)

{

if(\*p>='0'&&\*p<='9')

{

temp=\*p;

\*p=\*m;

\*m=temp;

p++;

m++;

}else

{

p++;

}

}

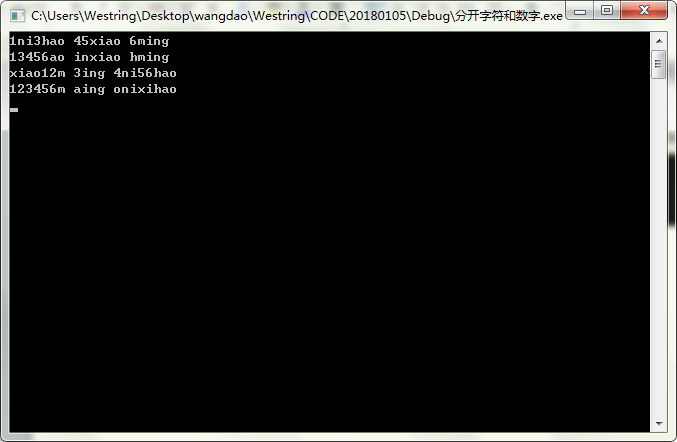
puts(a);

}

system("pause");

return 0;

}



1. 将 字 符 串 中 的 空 格 替 换 成 “%020” ， 例 如 “hello world how ”  
   ->”hello%020%020%020world%020%020%020how%020%020%020%020”

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

void ver(char a[100],char b[100])

{

char \*ap,\*bp;

ap=a+strlen(a)-1;

bp=b;

while(ap>=a)

{

if(\*ap==' ')

{

\*bp++='0';

\*bp++='2';

\*bp++='0';

\*bp++='%';

ap--;

}else

{

\*bp=\*ap;

bp++;

ap--;

}

\*bp='\0';

}

}

int main()

{

char a[100];

char b[100];

char \*bs,\*be;

char temp;

while(memset(a,0,sizeof(a)),gets(a)!=NULL)

{

ver(a,b);

puts(b);

bs=b;

be=b+strlen(b)-1;

while(be>bs)

{

temp=\*be;

\*be=\*bs;

\*bs=temp;

bs++;

be--;

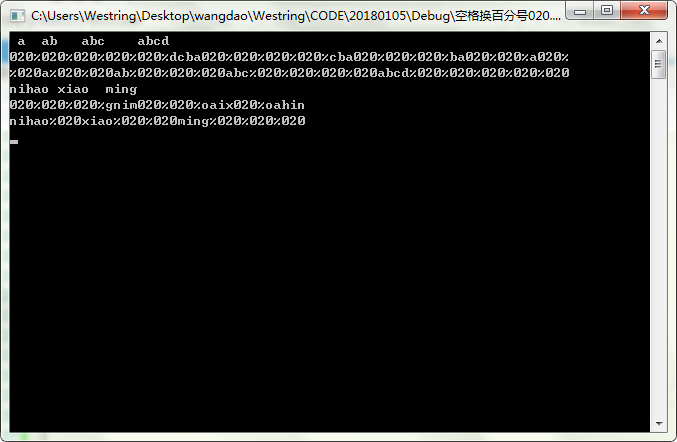
}

puts(b);

}

system("pause");

}



1. 删除字符串中指定的字符。 例如 “abcdaefaghiagkl“ 删除‘a’,以后： “bcdefghigkl”

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

int main()

{

char a[100];

char \*p,\*m;

while(memset(a,0,sizeof(a)),gets(a)!=NULL)

{

m=a;

while(\*m)

{

if(\*m=='a')

{

p=m;

while(\*p)

{

\*p=\*(p+1);

p++;

}

}else

{

m++;

}

}

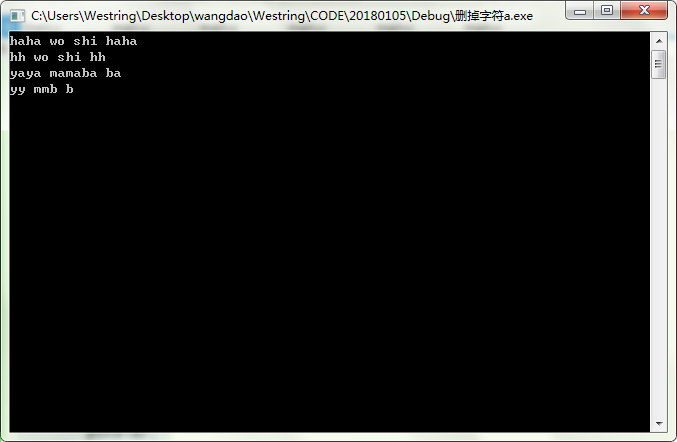
puts(a);

}

system("pause");

return 0;

}



1. 删除一个数组中重复的元素。例如 1 ,2, 2,2,3,3,3,4,4,5,5,5,6,6,6 -> 1,2,3,4,5,6

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

void xg\_quchong(char \*a)

{

char \*p,\*m;

p=a;

while(\*p)

{

m=a+strlen(a);

while(p<m)

{

if(\*p==\*m)

{

\*m=\*(m+1);

m--;

}else

{

m--;

}

}

p++;

}

}

int main()

{

char a[100];

while(memset(a,0,sizeof(a)),gets(a)!=NULL)

{

xg\_quchong(a);

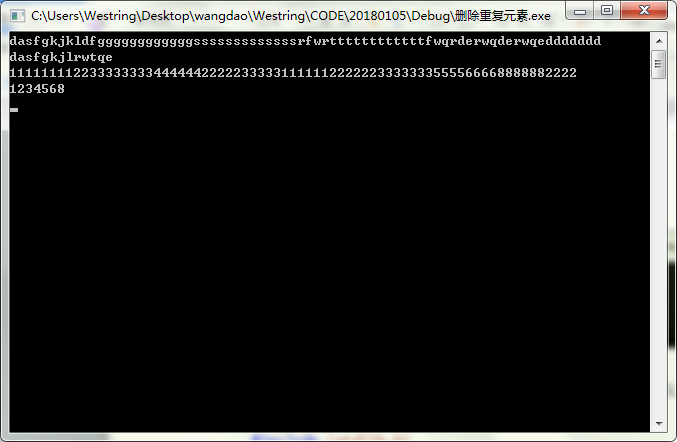
puts(a);

}

system("pause");

return 0;

}



5.. 将 字 符 串 中 的 相 邻 的 多 余 空 格 去 掉 ， 例 如 （空 格 用 下 划 线 表  
示）： ”\_\_\_hello\_\_\_\_world\_\_\_how\_are\_you\_\_” ->”hello\_world\_how\_are\_you”

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

void d\_space(char \*a)

{

char \*m,\*p;

m=a;

p=a;

while(\*m)

{

for(m=a;m<=a+strlen(a);m++)

{

if(\*m!=' ')

{

\*p=\*m;

p++;

}else if(\*m==' '&&\*(m+1)!=' ')

{

\*p=' ';

p++;

\*p=\*(m+1);

}

}

\*(p+1)='\0';

}

}

void d\_fspace(char \*a)

{

char \*m;

m=a;

if(\*a==' ')

{

while(\*m)

{

\*m=\*(m+1);

m++;

}

}

}

int main()

{

char a[100];

while(memset(a,0,sizeof(a)),gets(a)!=NULL)

{

d\_space(a);

puts(a);

d\_fspace(a);

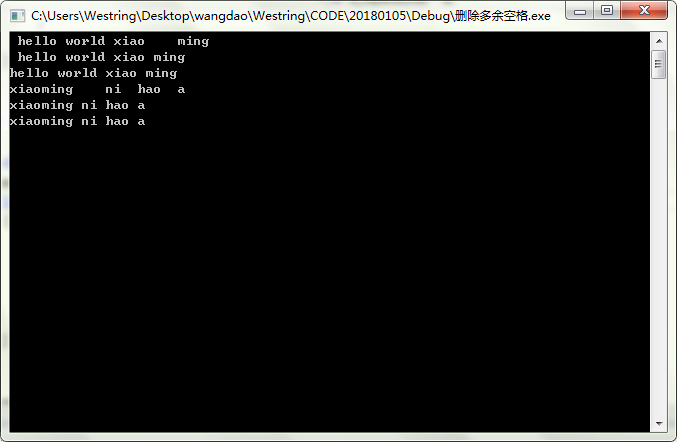
puts(a);

}

system("pause");

return 0;

}



6.附加题：大整数加法。 实现任意范围的两个整数的加法（ 整数的范围用 int 型的变量无法表示,50位）

7.求一个字符串数组的最大值和次大值 void big(char \*arr[],int size ,char\*\* big1,char\*\* big2)

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

void print(char \*p[])//这里可以写成 char \*\*p

{

int i;

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

{

puts(p[i]);

}

}

//二级指针的偏移，服务的是指针数组

int main()

{

char \*p[5];//定义了一个指针数组

char b[5][10]={"lele","lili","lilei","hanmeimei","zhousi"};

int i,j,tmp;

char \*t;

char \*\*p2;//定义一个二级指针

int a[5]={3,7,9,2,4};

for(i=0;i<5;i++)//让指针数组中的每一个指针都指向一个字符串

{

p[i]=b[i];

}

p2=p;

for(i=4;i>0;i--)//冒泡法排序

{

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

{

if(strcmp(p2[j],p2[j+1])==1)//判断 p2[j]是否大于 p2[j+1]

{

t=p2[j];

p2[j]=p2[j+1];

p2[j+1]=t;

}

}

}

print(p2);//先打印排序结果

//puts("----------------------------");

//for(i=0;i<5;i++)//再看数据存储本身有没有变

//{

// puts(b[i]);

//}

system("pause");

return 0;

}

