1. 统计该字符串中英文字母，数字，空格和其他字符的个数的算法

#include<stdio.h>

#include<Windows.h>

void main ()

{

char str[40]="Hhljfdsa 9988#\*vmncmv,a";

int cc=0,nn=0,ss=0,others=0,i;

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

{

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

cc++;

else if(str[i]>='0'&&str[i]<='9')

nn++;

else if(str[i]==' ')

ss++;

else

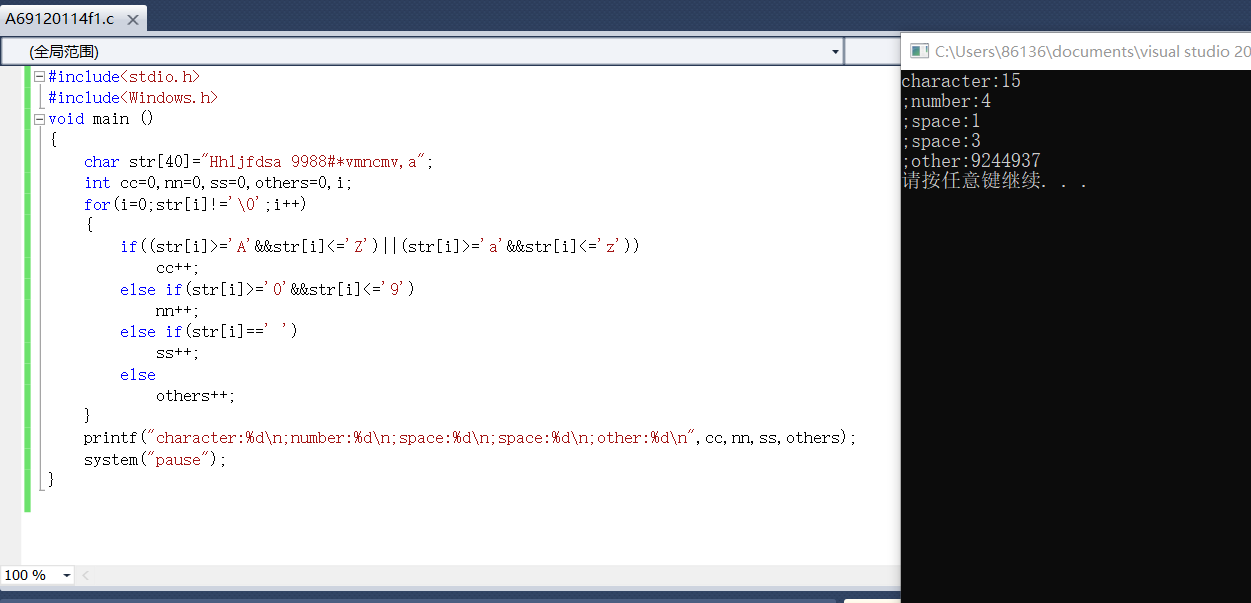
others++;

}

printf("character:%d\n;number:%d\n;space:%d\n;space:%d\n;other:%d\n",cc,nn,ss,others);

system("pause");

}



1. 起泡法对n个数进行从小到大的排序的算法

#include<stdio.h>

#include<Windows.h>

void bubble (int a[],int n)

{

int i,j,t;

for(i=1;i<n;i++)

for(j=n-1;j>=i;j--)

if(a[j]<a[j-1])

{

t=a[j];

a[j]=a[j-1];

a[j-1]=t;

}

}

void printArr(int a[],int n)

{

int i;

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

printf("%d",a[i]);

printf("\n");

}

void main()

{

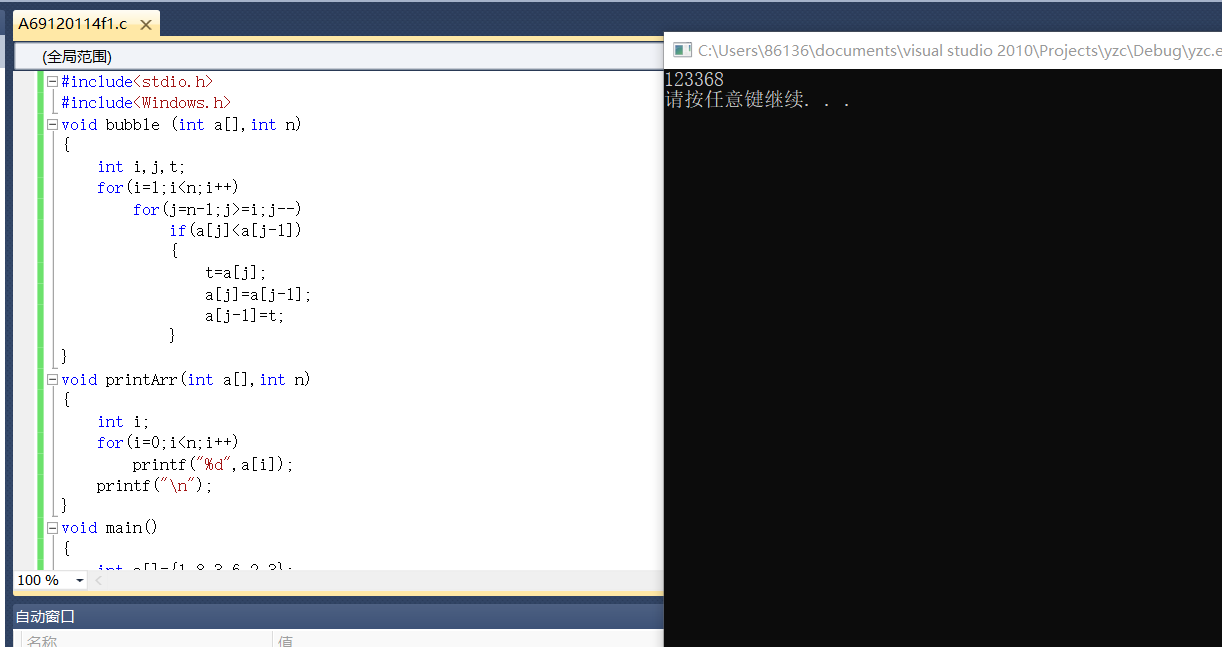
int a[]={1,8,3,6,2,3};

bubble(a,6);

printArr(a,6);

system("pause");

}



1. 选择排序法按从小到大的顺序排序的算法

#include<stdio.h>

#include<Windows.h>

void selectSort(int a[],int n)

{

int i,j,min,t;

for(i=0;i<n-1;i++)

{

min=i;

for(j=i+1;j<n;j++)

if(a[j]<a[min])

min=j;

if(min!=i)

{

t=a[min];

a[min]=a[i];

a[i]=t;

}

}

}

void printArr(int a[],int n)

{

int i;

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

printf("%d",a[i]);

printf("\n");

}

void main()

{

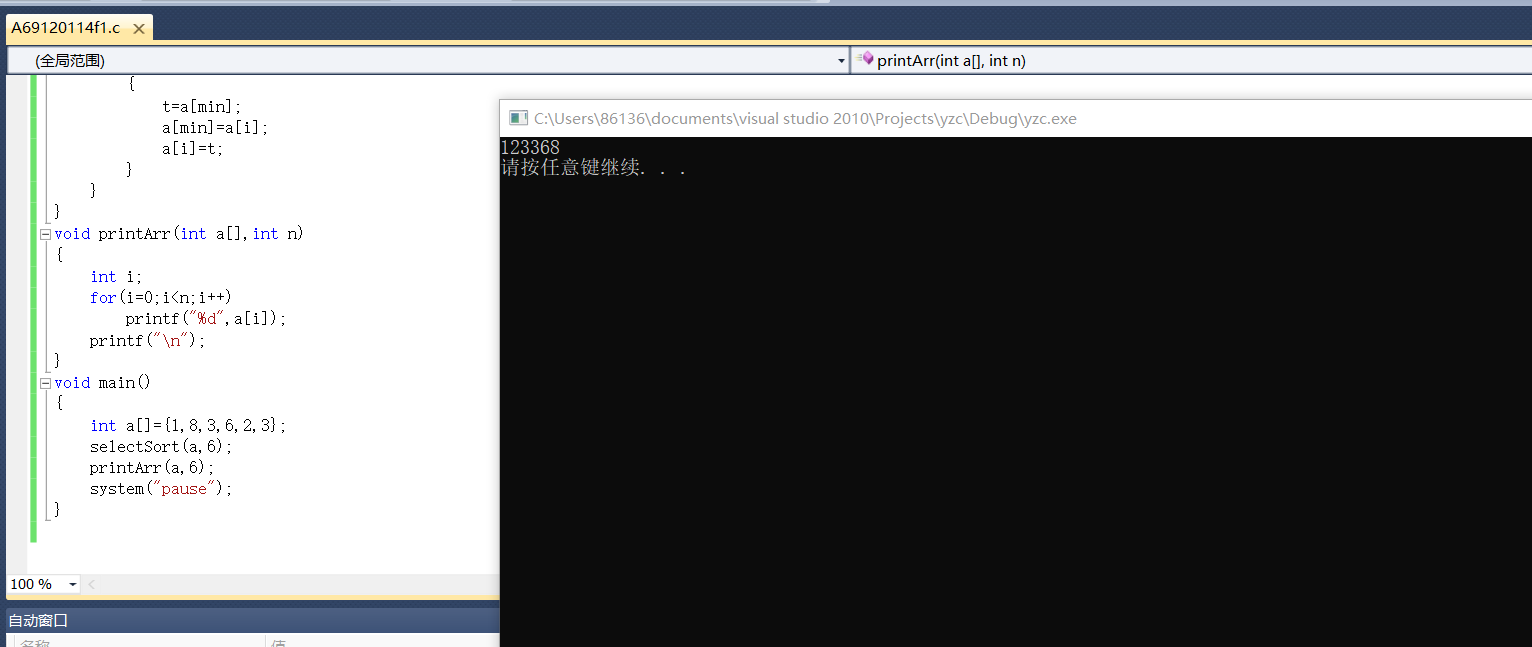
int a[]={1,8,3,6,2,3};

selectSort(a,6);

printArr(a,6);

system("pause");

}



1. 输入的字符串进行倒序处理的算法

#include<stdio.h>

#include<Windows.h>

void reverse(char str[])

{

char t;

int i,len;

len=strlen(str);

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

{

t=str[i];

str[i]=str[len-1-i];

str[len-1-i]=t;

}

}

void main()

{

char st[30];

printf("请?输º?入¨?一°?个?字Á?符¤?串ä?：êo");

gets(st);

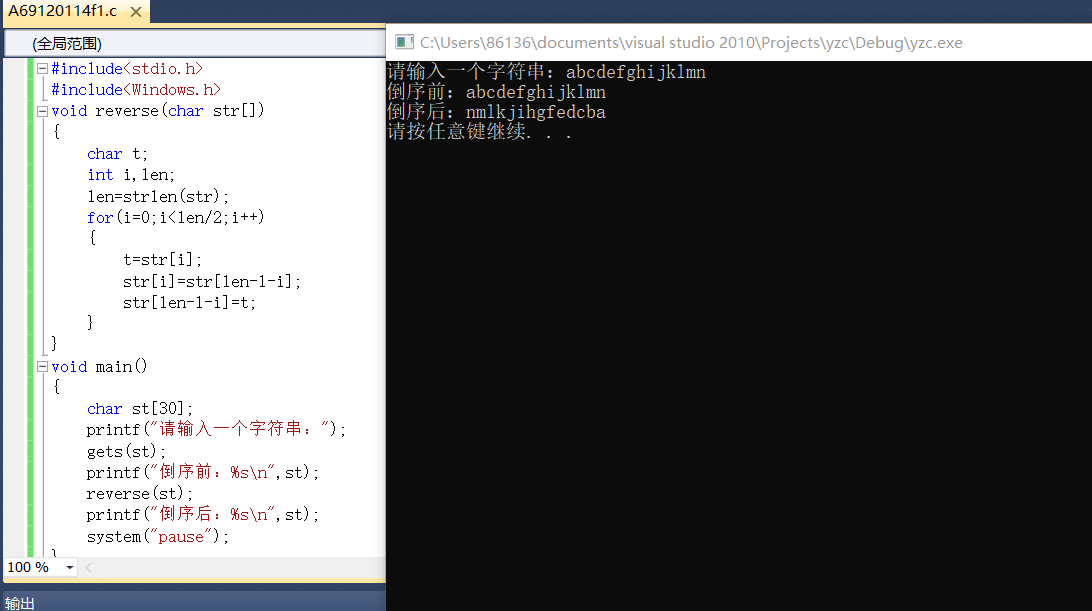
printf("倒Ì1序¨°前¡ã：êo%s\n",st);

reverse(st);

printf("倒Ì1序¨°后¨®：êo%s\n",st);

system("pause");

}



1. 字符数组1的全部字符复制到字符数组2的算法

#include<stdio.h>

#include<Windows.h>

void myStrcpy(char str1[],char str2[],int n)

{

int i;

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

str2[i]=str1[i];

}

void printArr(char str[],int n)

{

int i;

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

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

printf("\n");

}

void main()

{

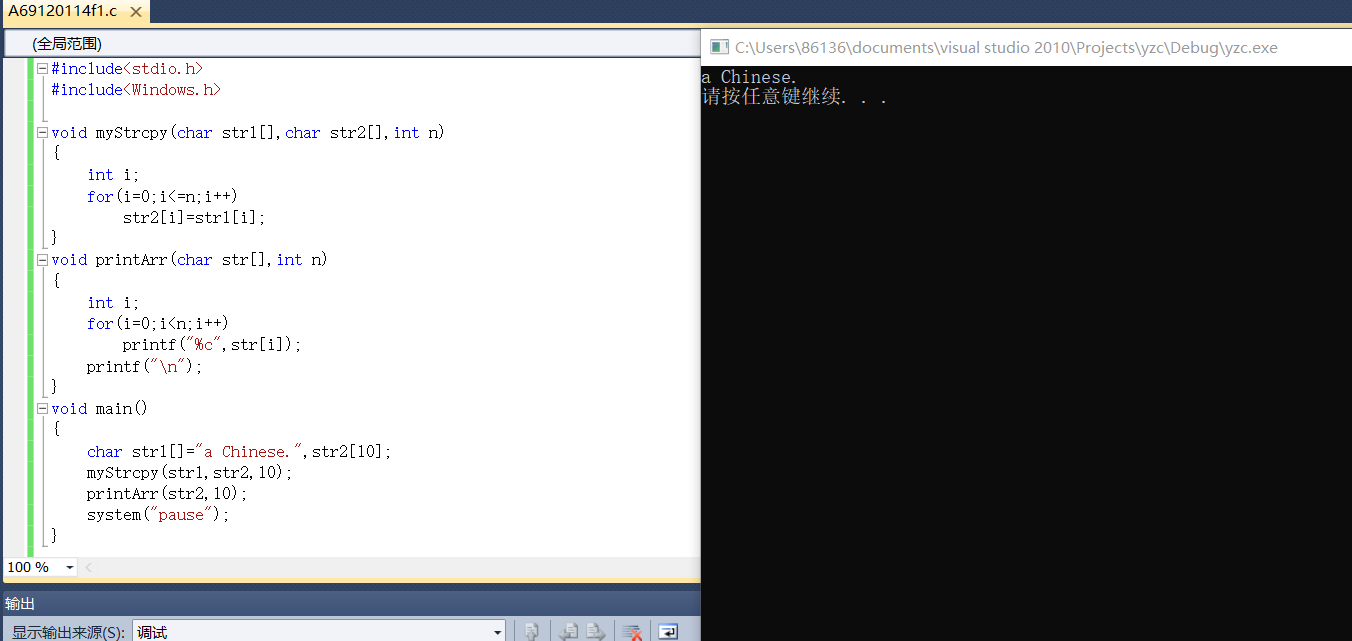
char str1[]="a Chinese.",str2[10];

myStrcpy(str1,str2,10);

printArr(str2,10);

system("pause");

}



1. 从二维整数组，求出最大值，最小值等的算法

#include<stdio.h>

#include<Windows.h>

void main ()

{

int a[3][4]={1,2,3,4,5,6,7,8,9,33,22,-1};

int max,min,sum=0,i,j;

max=a[0][0];

min=a[0][0];

for(i=0;i<3;i++)

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

{

if(max<a[i][j])

max=a[i][j];

if(min>a[i][j])

min=a[i][j];

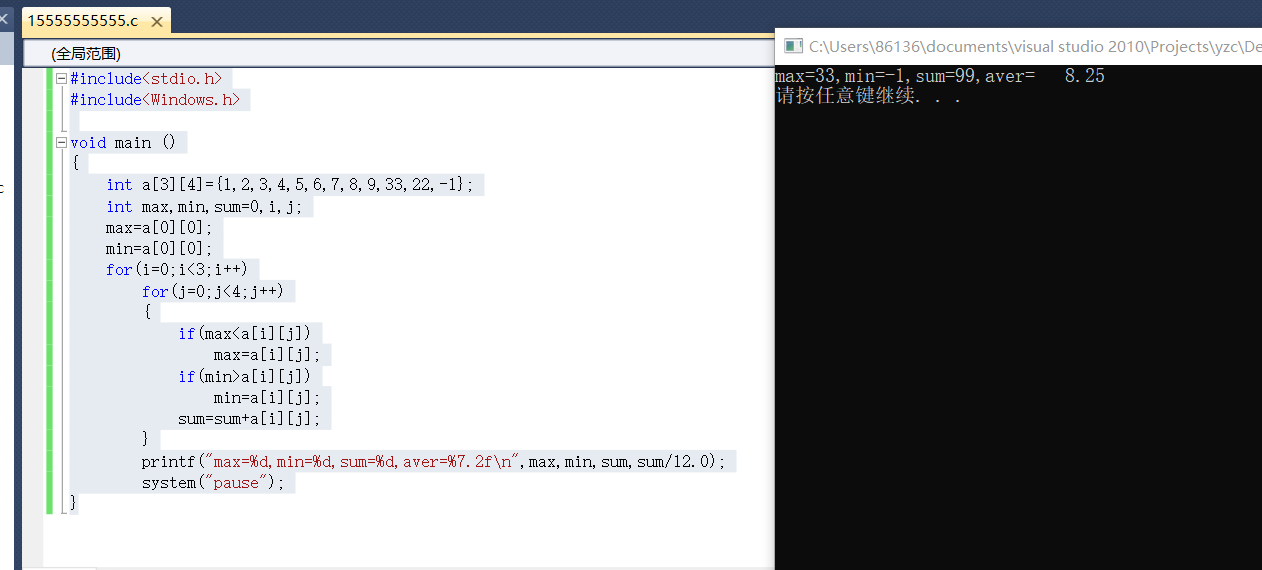
sum=sum+a[i][j];

}

printf("max=%d,min=%d,sum=%d,aver=%7.2f\n",max,min,sum,sum/12.0);

system("pause");

}



1. 从一维整型数组中找出的素数，并存入一另一个维一数组中的算法

#include<stdio.h>

#include<Windows.h>

int isPrime(int x)

{

int i;

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

if(x%i==0)

break;

if(i==x)

return 1;

else

return 0;

}

void main()

{

int a[8]={9,8,7,6,5,4,3,2},b[10];

int i,t;

t=0;

for(i=0;i<8;i++)

if(isPrime(a[i]))

{

b[t]=a[i];

t++;

}

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

printf("%d",b[i]);

printf("\n");

system("pause");

}



1. 从键盘输入整数的各元素，统计出正整数，负整数等的个数的算法

#include<stdio.h>

#include<Windows.h>

void main ()

{

int a,pos=0,neg=0;

scanf("%d",&a);

while(a!=0)

{

if(a>0)

pos++;

else

neg++;

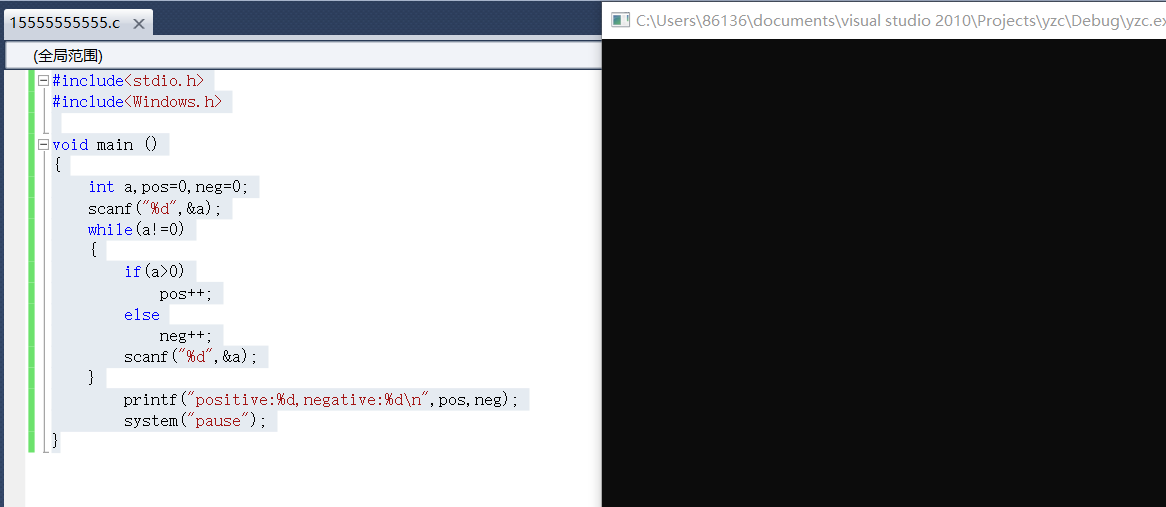
scanf("%d",&a);

}

printf("positive:%d,negative:%d\n",pos,neg);

system("pause");

}



1. 两个字符串进行连接的算法。

#include<stdio.h>

#include<Windows.h>

void myStrcat(char str1[],char str2[])

{

int i,len1,len2;

len1=strlen(str1);

len2=strlen(str2);

for(i=0;i<len2;i++)

str1[len1+i]=str2[i];

str1[len1+i]='\0';

}

void main()

{

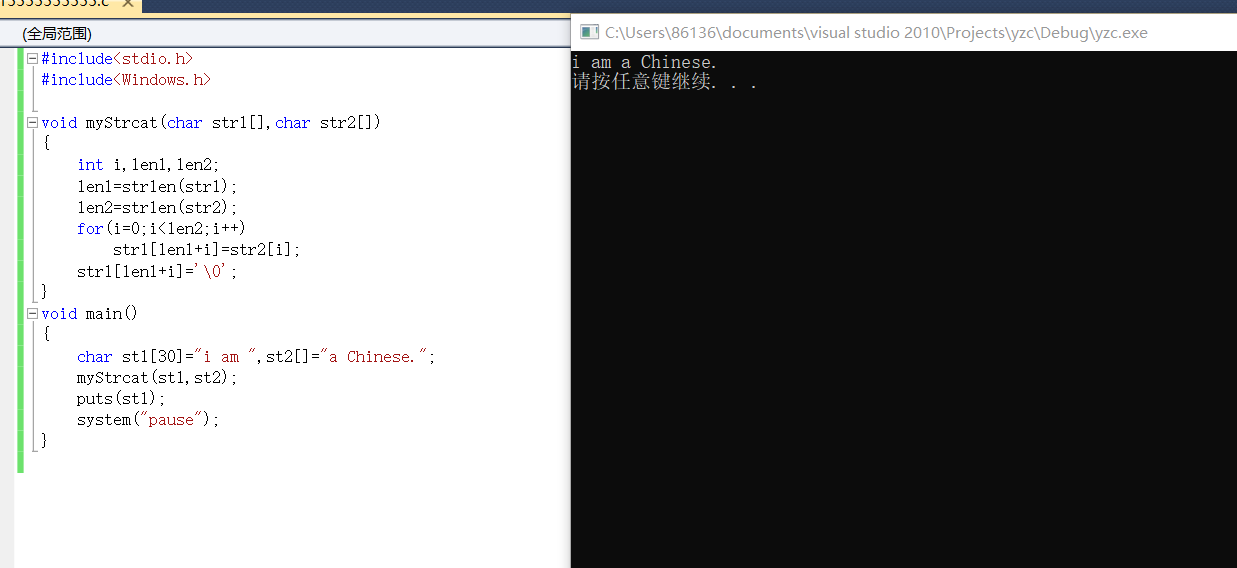
char st1[30]="i am ",st2[]="a Chinese.";

myStrcat(st1,st2);

puts(st1);

system("pause");

}



1. 将2个字符串合并后再转变为大写字母并输出的算法

#include<stdio.h>

#include<string.h>

#include<Windows.h>

void mergeTwo(char str1[],char str2[])

{

int len1,len2,i;

len1=strlen(str1);

len2=strlen(str2);

for(i=0;i<len2;i++)

str1[len1+i]=str2[i];

str1[len1+i]='\0';

}

void toUpper(char str[])

{

int i=0;

while(str[i]!='\0')

{

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

str[i]=32;

i++;

}

}

void main()

{

char st1[30]="fjajds333HHh",st2[]="dfaf";

mergeTwo(st1,st2);

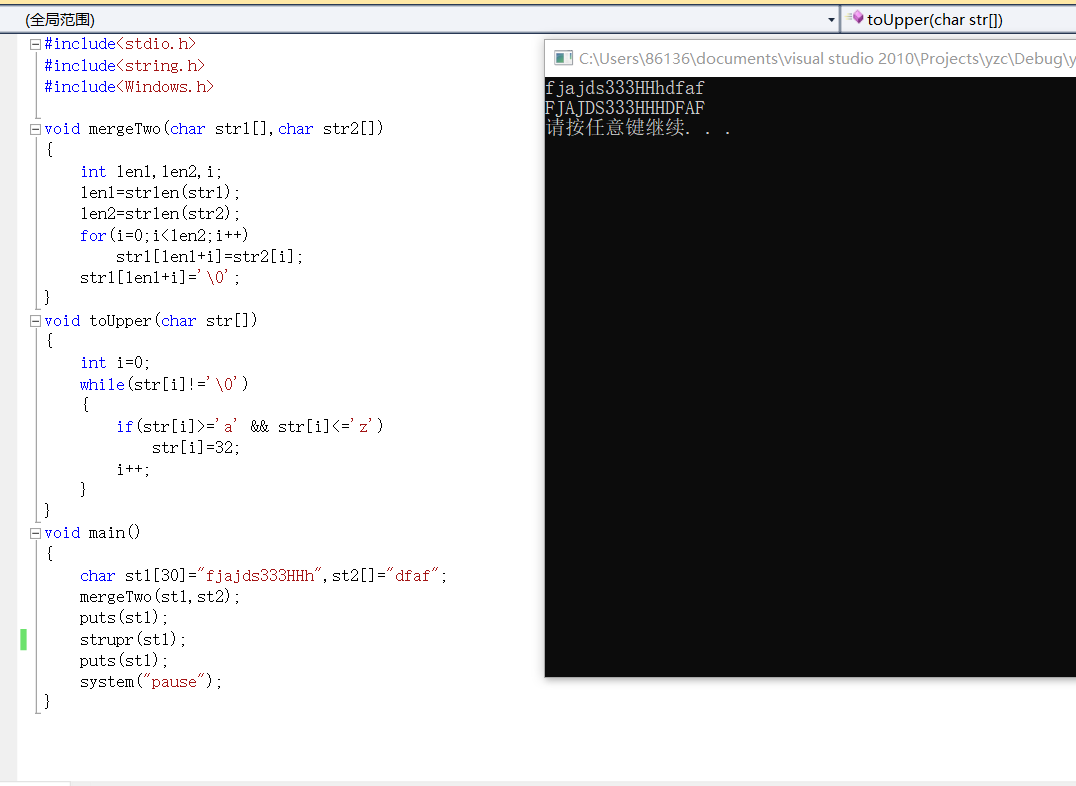
puts(st1);

strupr(st1);

puts(st1);

system("pause");

}



1. 顺序查找法从n个元素的数组中，查找某个数据的算法

#include<stdio.h>

#include<Windows.h>

int find(int a[],int n,int data)

{

int found=-1;

int i;

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

if(a[i]==data)

{

found=i;

break;

}

return found;

}

void main()

{

int a[]={1,2,3,4,5,6,7};

int res;

res=find(a,7,3);

if(res<0)

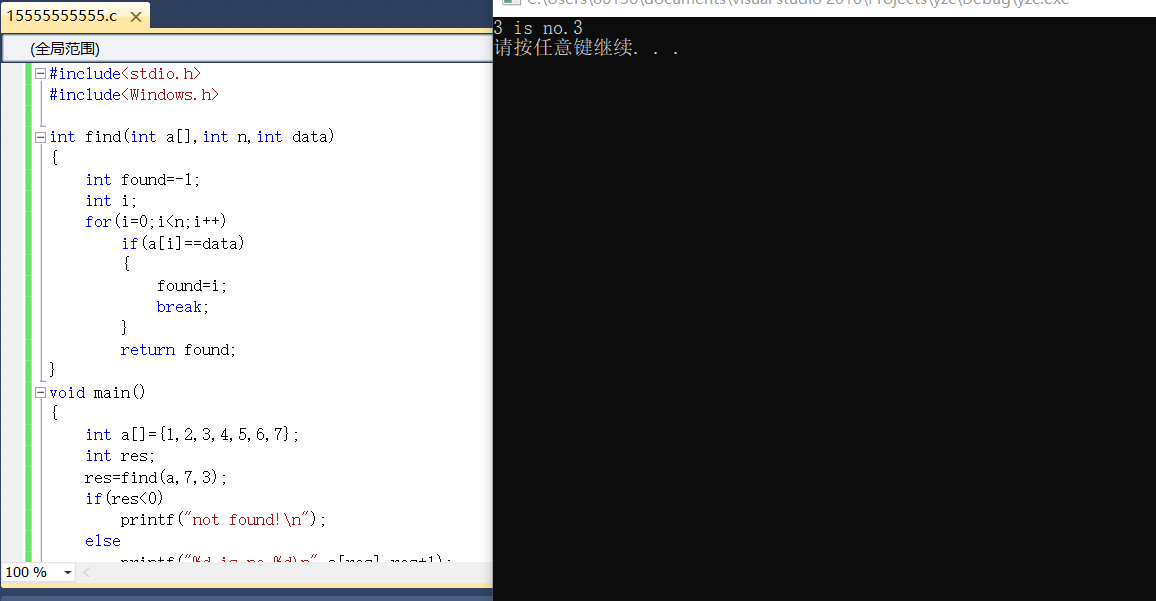
printf("not found!\n");

else

printf("%d is no.%d\n",a[res],res+1);

system("pause");

}



1. 素数的算法

#include<stdio.h>

#include<Windows.h>

int isPrime(int x)

{

int i;

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

if(x%i==0)

break;

if(i==x)

return 1;

else

return 0;

}

void main()

{

int a;

scanf("%d",&a);

if(isPrime(a))

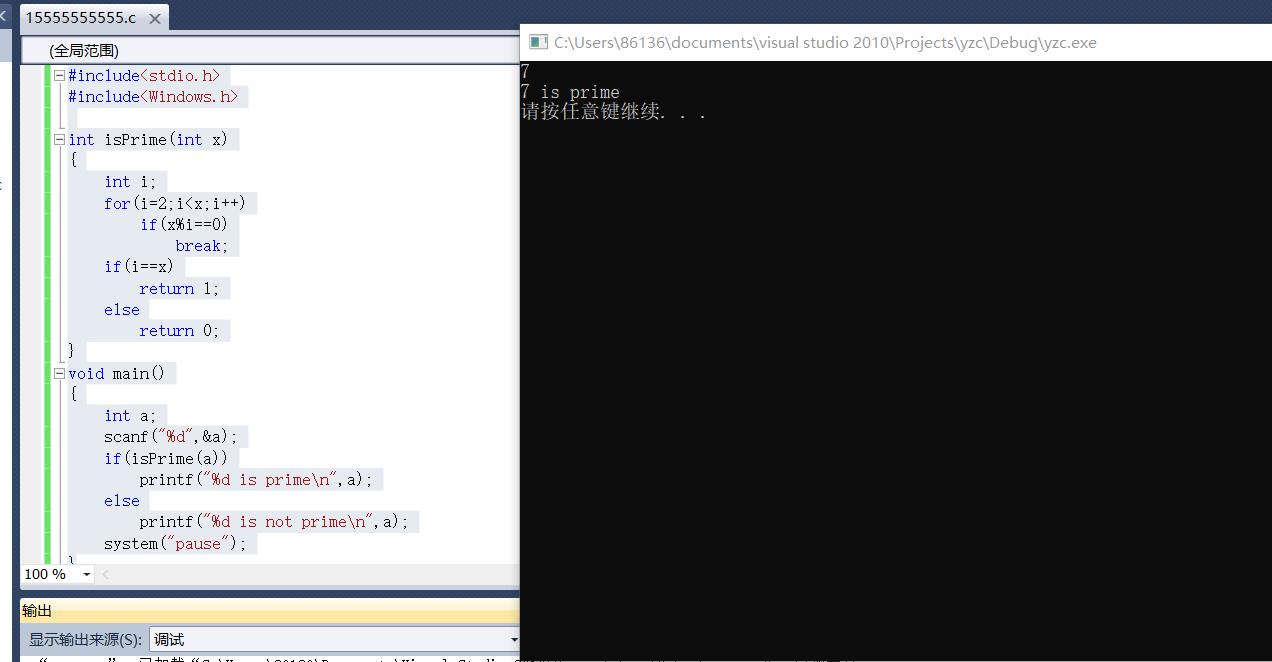
printf("%d is prime\n",a);

else

printf("%d is not prime\n",a);

system("pause");

}



1. 闰年的算法

#include<stdio.h>

#include<Windows.h>

int isLeap(int year)

{

return year%4==0&&year%100!=0||year%400==0;

}

void main()

{

int a;

scanf("%d",&a);

if(isLeap(a))

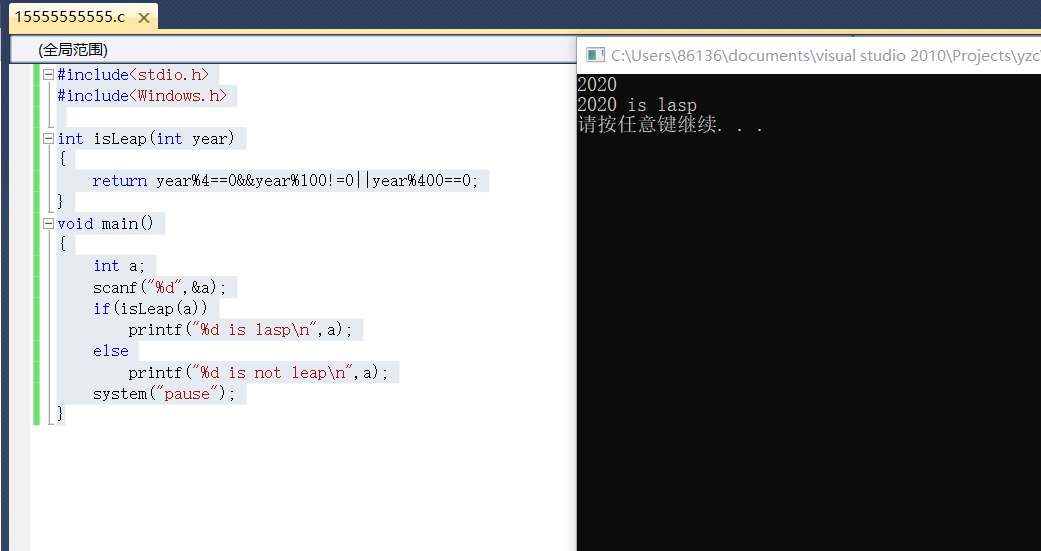
printf("%d is lasp\n",a);

else

printf("%d is not leap\n",a);

system("pause");

}



1. 在一个字符串中，计算有几个单词的算法

#include<stdio.h>

#include<Windows.h>

int countWord(char str[])

{

int i,word=0,count=0;

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

{

if(str[i]==' ')

word=0;

else if(word==0)

{

count++;

word++;

}

}

return count;

}

void main()

{

char str[]="i am a chinese.";

int t;

t=countWord(str);

printf("%d\n",t);

system("pause");

}

