目录

[复习 1](#_Toc26541458)

[思考1：简单指针 1](#_Toc26541459)

[思考2：指针和数组 1](#_Toc26541460)

[指针和串 1](#_Toc26541461)

[指针遍历串的方法 1](#_Toc26541462)

[上述改进，提高 2](#_Toc26541463)

[利用指针实现两个串的连接，不用函数 3](#_Toc26541464)

[3、使用指针，部分改造 4](#_Toc26541465)

[4、标准指针实现 5](#_Toc26541466)

[5、知识点补充 6](#_Toc26541467)

[作业： 6](#_Toc26541468)

## 复习

### 思考1：简单指针

#include<stdio.h>

int main()

{int x,\*p;

p=&x;

\*p=3;

printf("%d,%d",\*p,\*&x);//3,3

}

### 思考2：指针和数组

#include<stdio.h>

int main()

{int i,a[5]={3,5,7,12,5},\*p;

p=a;//a 是数组名，挪用为数组的首地址

for(i=0;i<5;i++)//也可以认为,\*(a+i)<===>a[i]

printf("%4d",\*(a+i));//地址本来没有数学运算，但加减代表其他

printf("\n");

for(i=0;i<5;i++)

printf("%4d",\*(p+i));

printf("\n");

for(i=0;i<5;i++)

printf("%4d",p[i]);//上述三种的输出效果一样

printf("\n");

}

## 指针和串

### 指针遍历串的方法

#include<stdio.h>

int main()

{int i,a[5]={3,5,7,12,5},\*p;

p=a;//a 是数组名，挪用为数组的首地址

for(i=0;i<5;i++)//也可以认为,\*(a+i)<===>a[i]

printf("%4d",\*(a+i));//地址本来没有数学运算，但加减代表其他

printf("\n");

for(i=0;i<5;i++)

printf("%4d",\*(p+i));

printf("\n");

for(i=0;i<5;i++)

printf("%4d",p[i]);//上述三种的输出效果一样

printf("\n");

}

### 上述改进，提高

#include<stdio.h>

int main()

{char str[200],\*p;

/\*int a[5]={3,4,5};

char str[200]={"china"}

str[200]={'c','h','i','n','a','\0','\0','\0'...补满}

\*/

int i=0;//统计串符号的个数 不能用strlen

gets(str);//"china"

p=str;

while(\*p!='\0') //切记，'\0'的ASCII码是0

//上一行等价于while(\*p!=0)

//难度：while(\*p)

/\*

if (x!=0)<==>if (x)

while(x!=0)<<====>> while(x)

\*/

{i++;

p++;

}

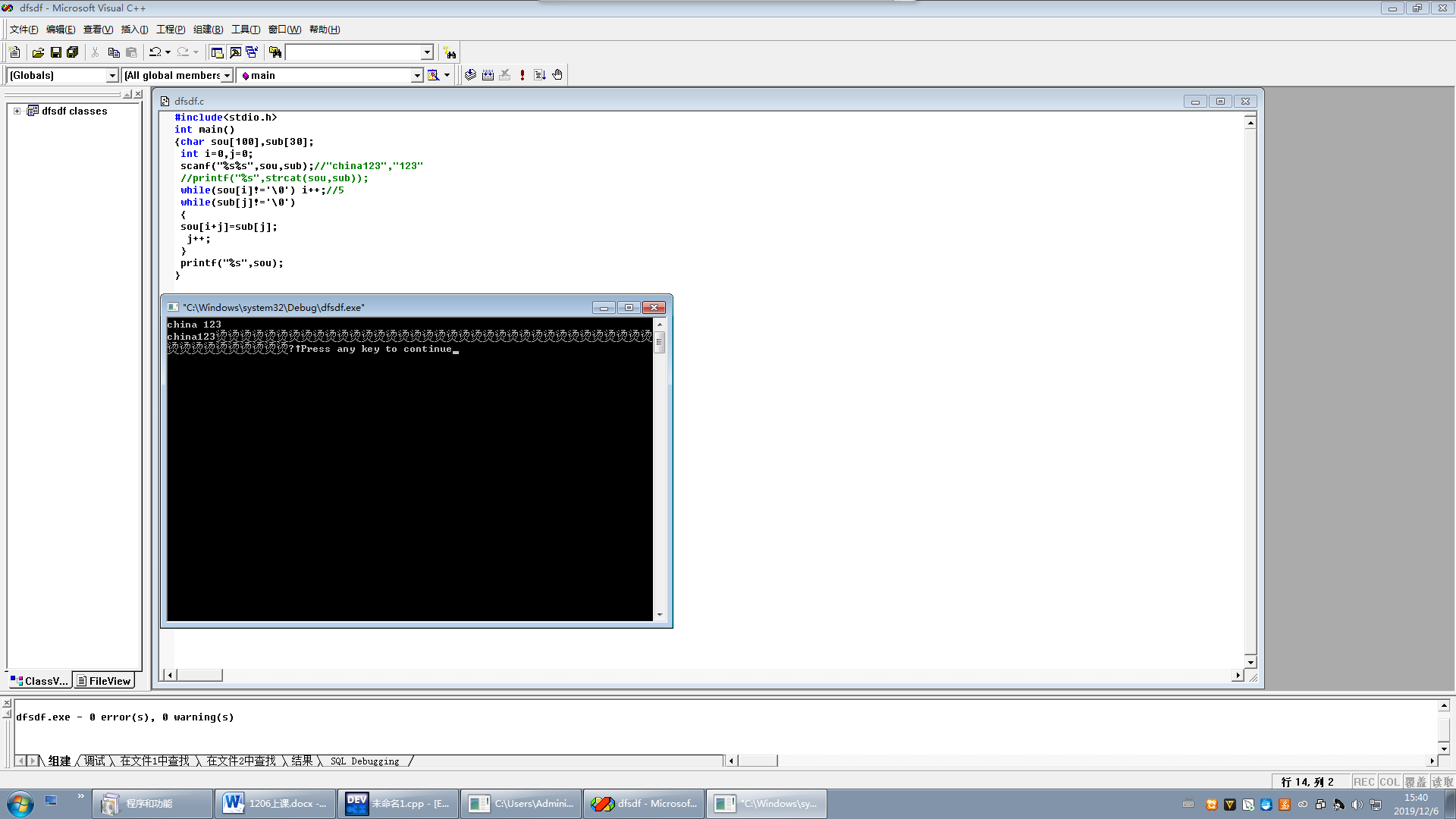
\*/

printf("%c,%d",'\0','\0');

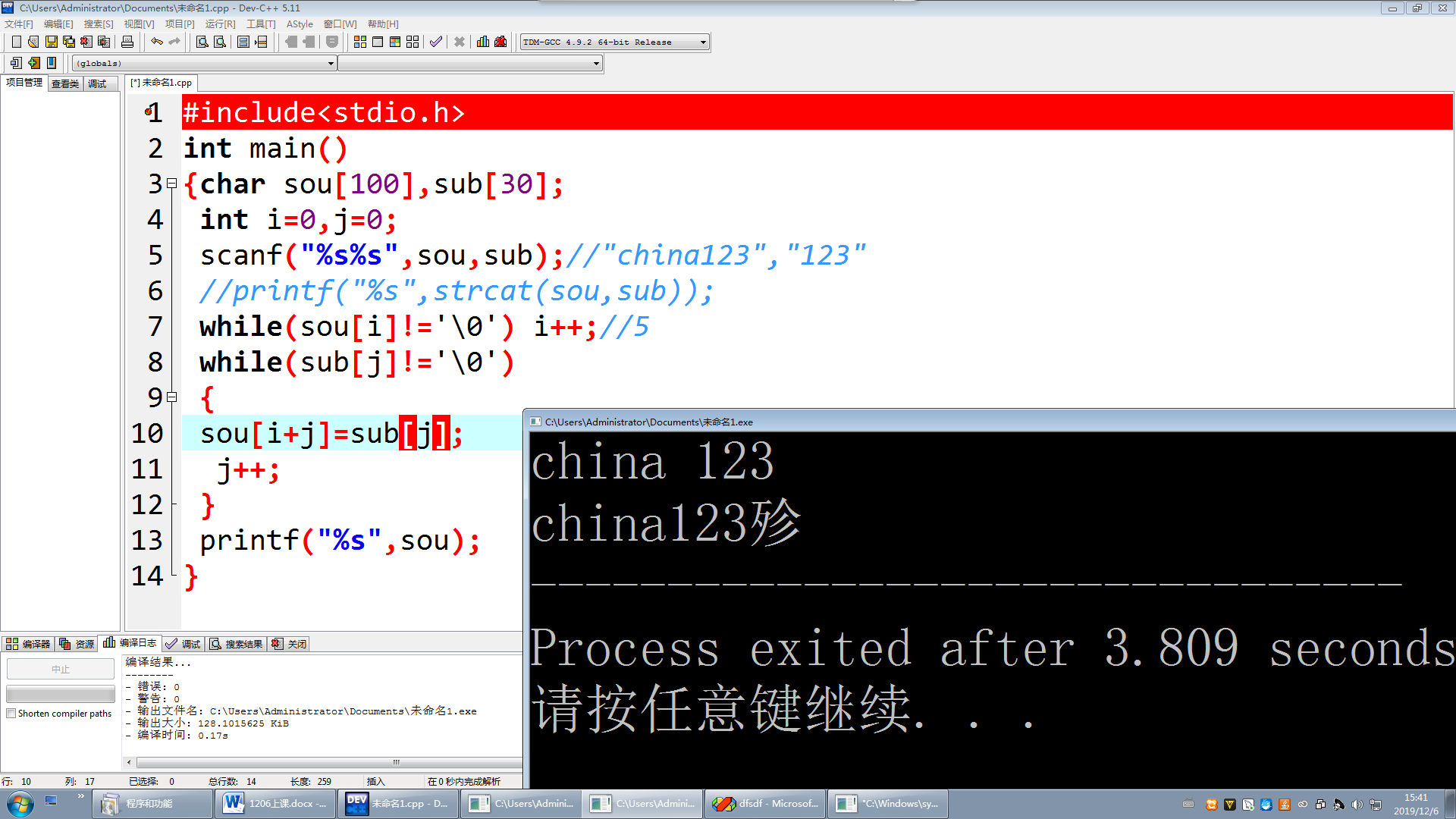
}

### 利用指针实现两个串的连接，不用函数

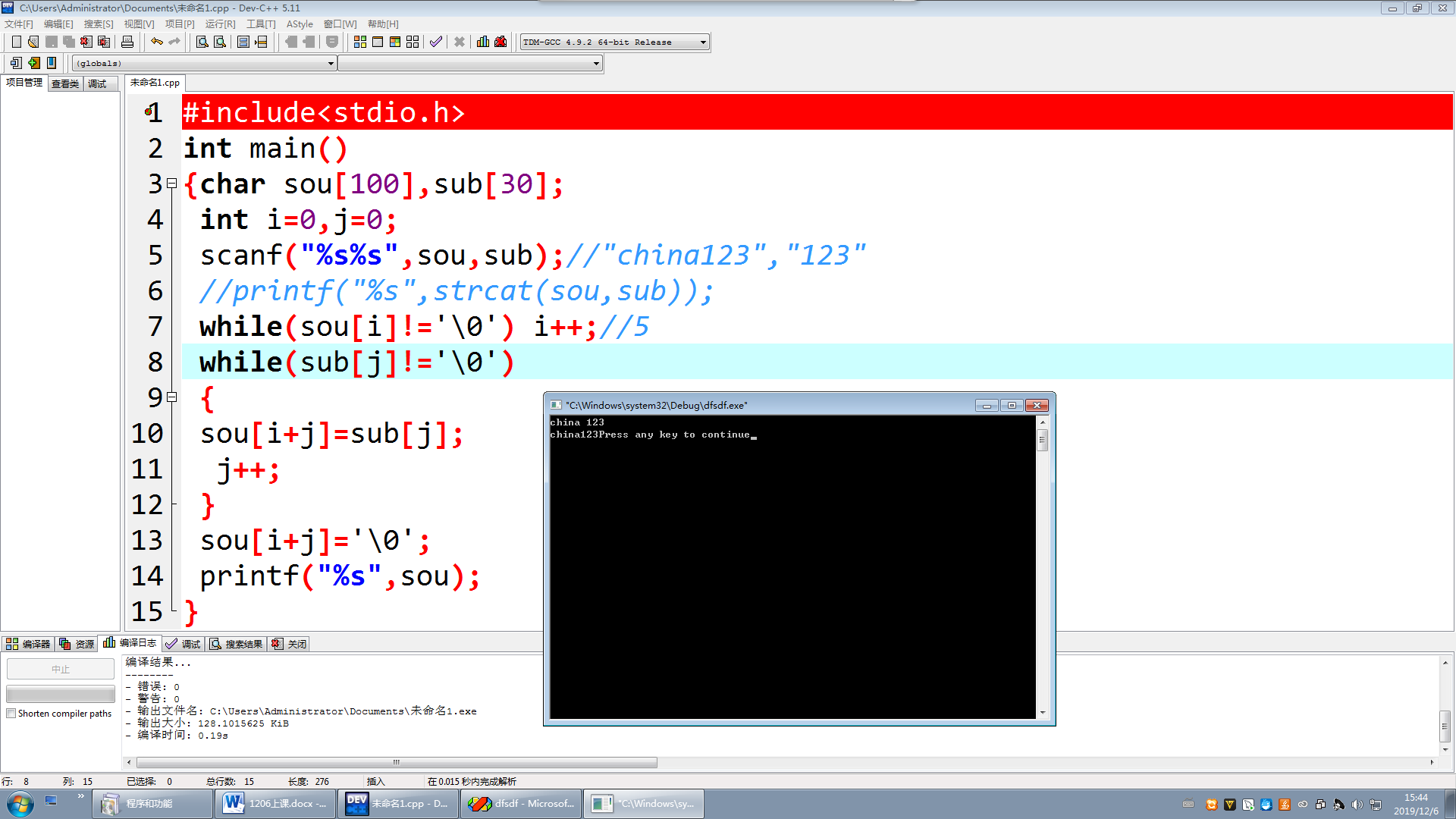
#### 不用指针，错误情况1



#### 错误情况2



#### 上述改进如下：



### 使用指针，部分改造

#include<stdio.h>

int main()

{char sou[100],sub[30];

int i=0,j=0;

scanf("%s%s",sou,sub);//"china123","123"

//printf("%s",strcat(sou,sub));

while(\*(sou+i)!='\0') i++;//只改造部分，其他参照

while(\*(sub+j)!='\0')

{

sou[i+j]=sub[j];

j++;

}

sou[i+j]='\0';

printf("%s",sou);

}

### 标准指针实现

#include<stdio.h>

int main()

{char sou[100],sub[30],\*p1,\*p2;

p1=sou;p2=sub;

scanf("%s%s",sou,sub);//"china","123"

while(\*p1!='\0') p1++;

while(\*p2!='\0')

{\*p1=\*p2;

p1++;p2++;

}

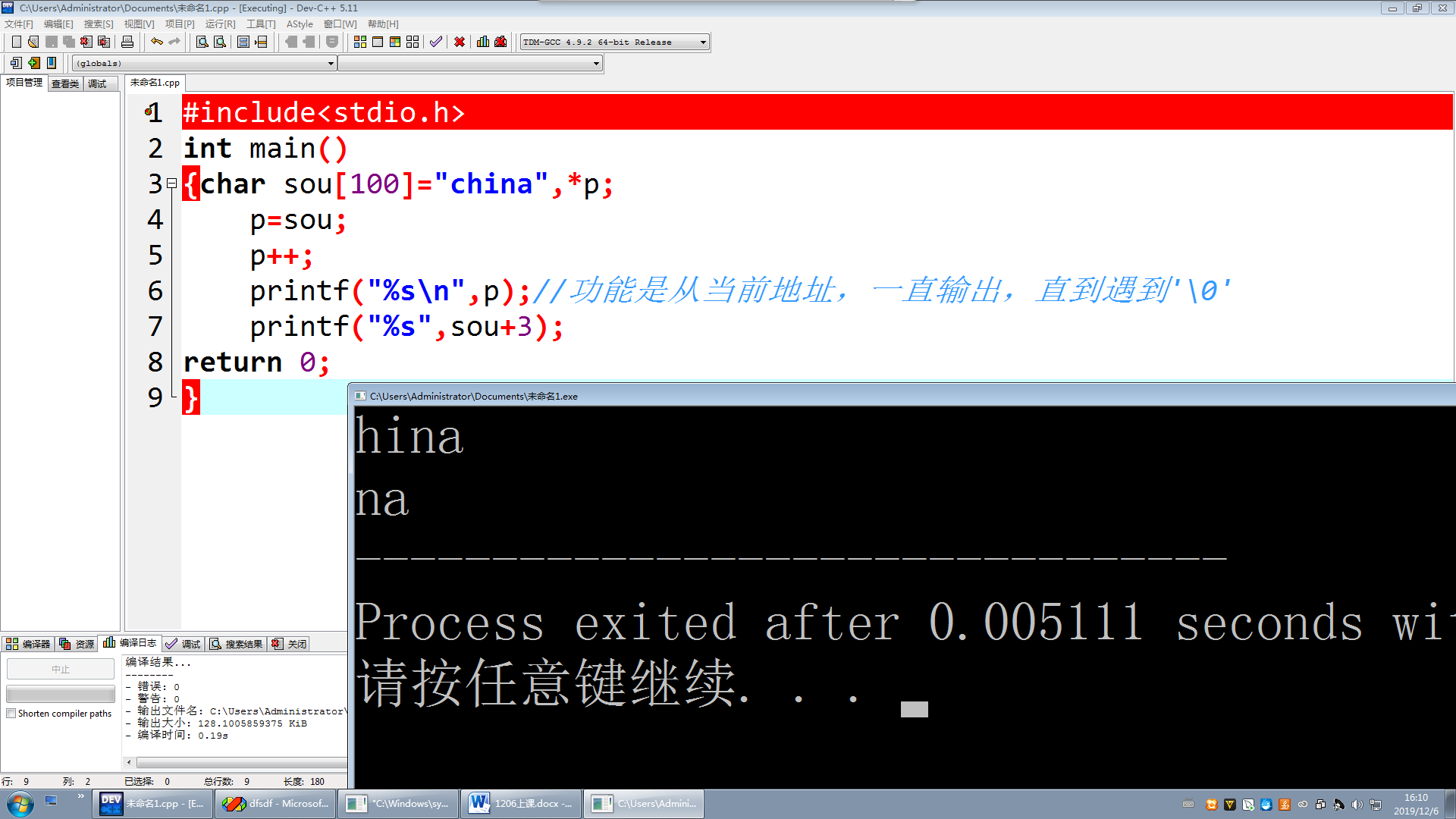
\*p1='\0';

printf("%s",sou); //不能输出p1

return 0;

}

### 知识点补充



### 作业：

1、利用指针，找出一维整数中的最大值

2、利用指针，手工实现两个串的连接。

提醒：交作业时间，下周一上午之内；建议大家交作业时将作业翻到答题的起始页。

另外，以后每周二下午在图书馆3楼08号机房上机。