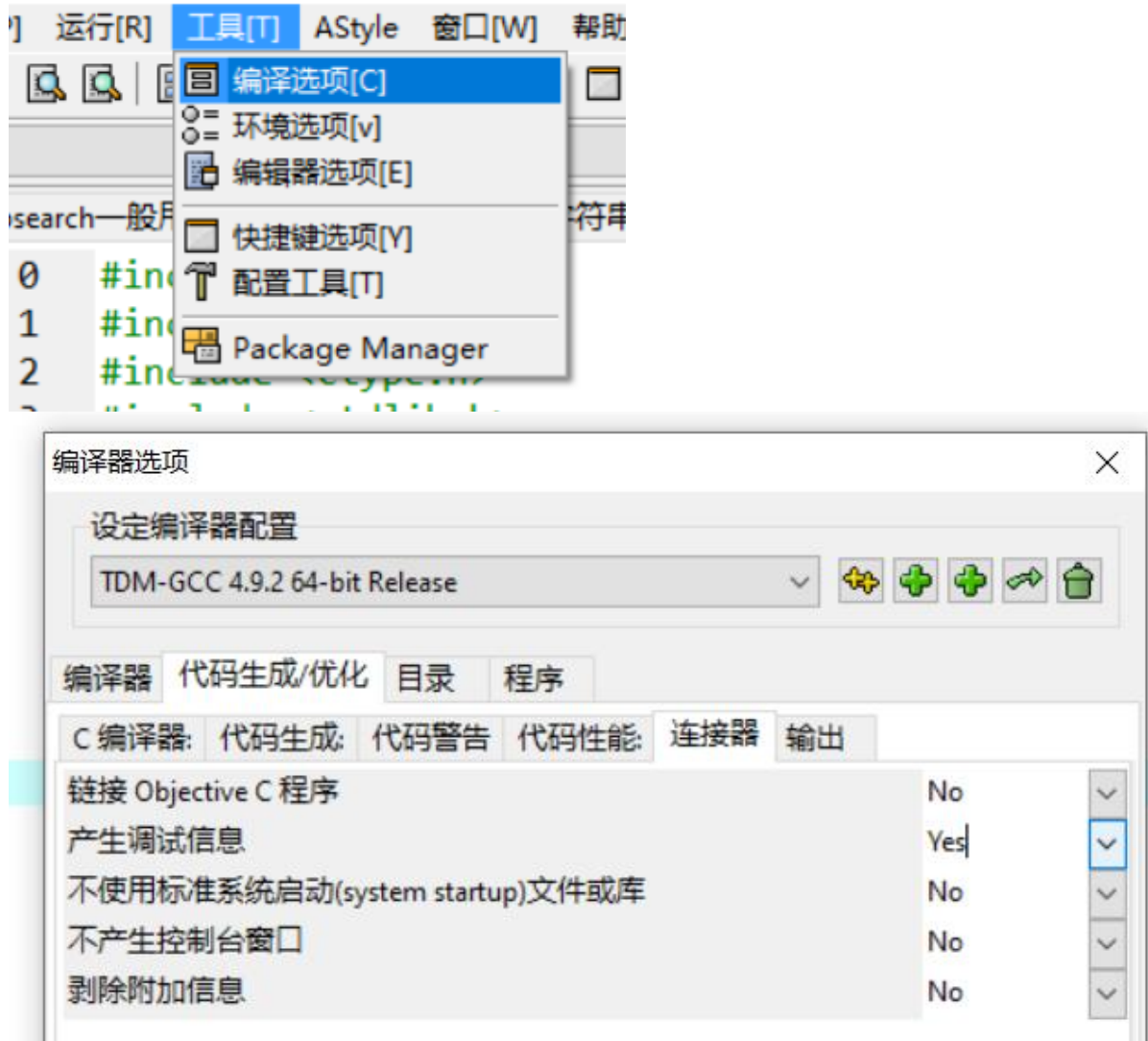


## Dev C++基本调试方法（主流版本应该都可行）

### 基本设置

1.工具 -> 编译选项 -> 代码生成/优化 -> 连接器 -> 产生调试信息，设为Yes，确定



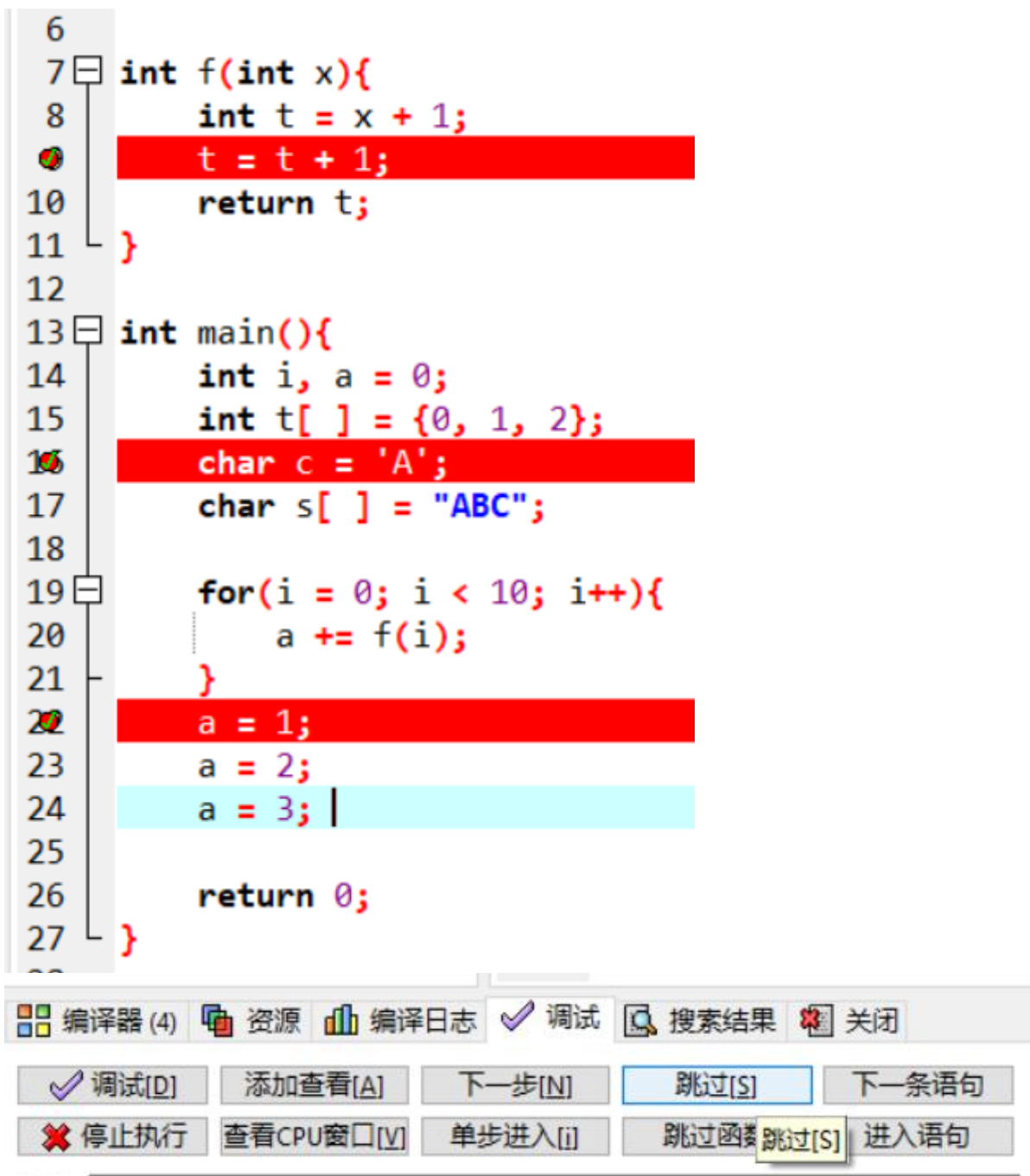
### 建议使用输入输出重定向

- 1.输入最好用freopen语句，写在int main() 下一行，从指定文件输入数据。除非你的程序不需要输入数据（比如后面的例子）
- 2.输出可以不用重定向，但是如果输出非常庞大，而你需要反复查看，可以输出到指定的文件中。如果你指定的这个文件不存在，系统会自动创建相同名称的文件
- 3.freopen里的文件名必须带后缀，且必须与实际的文件名相同。指向的文件必须与你运行的.c文件在同一个文件夹里
- 4.必须确保系统不隐藏后缀名。如果你考试用的电脑隐藏了后缀名而你不知道怎么办，建议呼叫老师或助教帮你

```
13  □ int main(){  
14      freopen("in.txt", "r" , stdin);  
15  
16      .....  
17  
18      freopen("out.txt", "w", stdout);  
19      return 0;
```

#### 添加断点、跳过断点

- 1.在旁边的行数栏点一下就可以把这行设为断点
- 2.调试时至少需要有一个断点，调试将从第一个断点开始。如果没有断点，调试直接结束
- 3.如果要从一个断点直接到下一个断点，可以点下面调试栏里的“跳过”
- 4.其它函数里也可以设置断点，如果主程序要用到这个函数，当你跳到下一个断点时就会在这里停下
- 5.断点的先后是从main函数开始为先，而不是写在上面或先设置为先。比如下图第16、22行的断点就先于第9行



### 添加查看、移除查看

- 1.至少设置一个断点后，点左下角的“调试”就可以开始调试了。开始调试时，蓝色行标志当前调试到哪一行，查看的变量也更新到这一行的前一行（这行暂不执行）
- 2.点下方的“添加查看”可以添加各种想查看的数据，形式也不唯一。注意查看数组元素时不会检查下标越界。比如下图，t[4]就是越界了，但是没有报错。
- 3.注意，当调试进入一个新的函数，如果恰好有变量名相同，编译器会显示新的函数里变量的值
- 4.全局变量千万不要与局部变量名称相同
- 5.在左边列表右键单击可以移除一个或全部变量

编译器 (5) 资源 编译日志 调试 搜索结果 关闭

调试[D] 添加查看[A] 下一步[N] 跳过[S] 下一条语句  
停止执行 查看CPU窗口[V] 单步进入[i] 跳过函数 进入语句

i = 0  
a = 0  
t = {0, 1, 2}  
t[1] = 1  
t[4] = 0  
c = 65 'A'  
s = "ABC"  
s[t[1]] = 66 'B'

```
5  
6  
7 int f(int x){  
8     int t = x + 1;  
9     t = t + 1;  
10    return t;  
11 }  
12  
13 int main(){  
14     int i, j, a = 0;  
15     int t[ ] = {0, 1, 2};  
16     char c = 'A';  
17     char s[ ] = "ABC";  
18  
19     for(i = 0; i < 10; i++){  
20         a += f(i);  
21     }  
22     a = 1;  
23     a = 2;  
24     a = 3;  
25  
26     return 0;
```

i = 0  
a = 0  
t = {0, 1, 2}  
t[1] = 1  
t[4] = 0  
c = 65 'A'  
s = "ABC"  
s[t[1]] = 66 'B'

添加查看[A]  
修改数据[M]  
移除查看[R]  
全部清除[C]

```
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15
```

## 下一步、单步进入

1.通常要调试往后走，只用“下一步”和“单步进入”就够了，其它的少用

2.“下一步”遇到函数时会直接执行（不论是自己写的还是函数库的），不进入详细步骤。此时点“单步进入”可以进入函数内部。如图，就进入了f的内部

```
7  int f(int x){
8      int t = x + 1;
9      t = t + 1;
10     return t;
11 }
12
13 int main(){
14     int i, j, a = 0;
15     int t[ ] = {0, 1, 2};
16     char c = 'A';
17     char s[ ] = "ABC";
18
19     for(i = 0; i < 10; i++){
20         a += f(i);
21     }
```

```
7  int f(int x){
8      int t = x + 1;
9      t = t + 1;
10     return t;
11 }
12
13 int main(){
14     int i, j, a = 0;
15     int t[ ] = {0, 1, 2};
16     char c = 'A';
17     char s[ ] = "ABC";
18
19     for(i = 0; i < 10; i++){
20         a += f(i);
21     }
```

## 查看输出

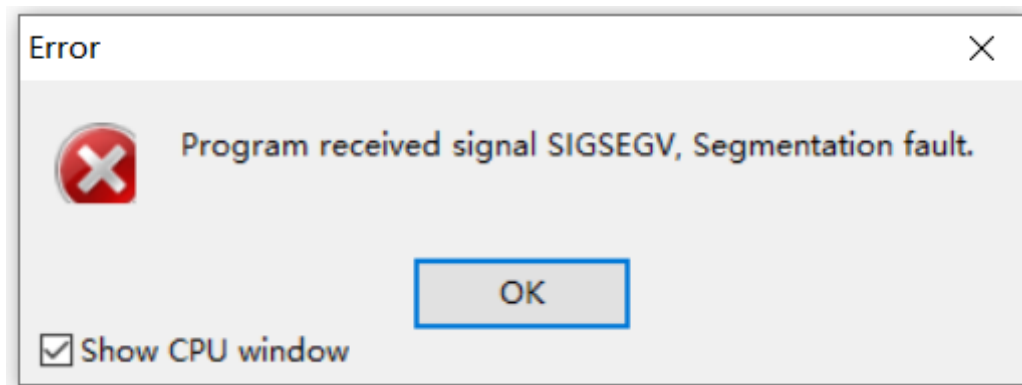
调试开始时也会产生一个输出窗口，可以看到目前已经执行的输出结果

## 结束调试

点左下角停止执行就可以了

## 其它注意

1.有时会出这样的问题，这说明可能出了这样的问题：格式不对，指针操作错误、内存不够、数组越界、递归层数太大等等



2.Dev C++的调试有时会出问题，如果你发现调试过程中的数据很奇怪，建议多调试几次或者改为中间输出一些数据看看

3.还有一些小细节，受限于能力经验不足，也一时想不周全，可能有所遗漏。所以大家还是多拿几个程序试试，很快就有门道了