# 第二次上机

### 实验内容

编写一个多文件程序，完成如下任务：

1.从键盘上随机输入5个整数（0-100）；

2.寻找5个整数中的最大值和最小值并显示；

要求：

1.在主程序中完成5个数的输入；

2.求最大值函数和最小值函数要分别用一个文件；

### 二．程序代码

#include <iostream>

using namespace std;

int max(int x);

int min(int x);

int main()

{

int x, y,c, a[5], m = 100, n = 0, t = 0, q = 0;

cin >> x;

while (t < 4)

{

a[t] = x;

m = min( x);

n = max( x);

cin >> x;

t++;

}

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

{

if ((a[c] > 0 && a[c] < 100) || (a[c] == 0) || (a[c] == 100))

q++;

}

if (q == 4)

cout << "min=" << m << endl << "max=" << n << endl;

else

cout << "已超出范围" << endl;

return 0;

}

int max(int x)

{

static int a = 0;

if (x > a)

{

a = x;

}

return a;

}

int min(int x)

{

static int a = 101;

if (x < a)

{

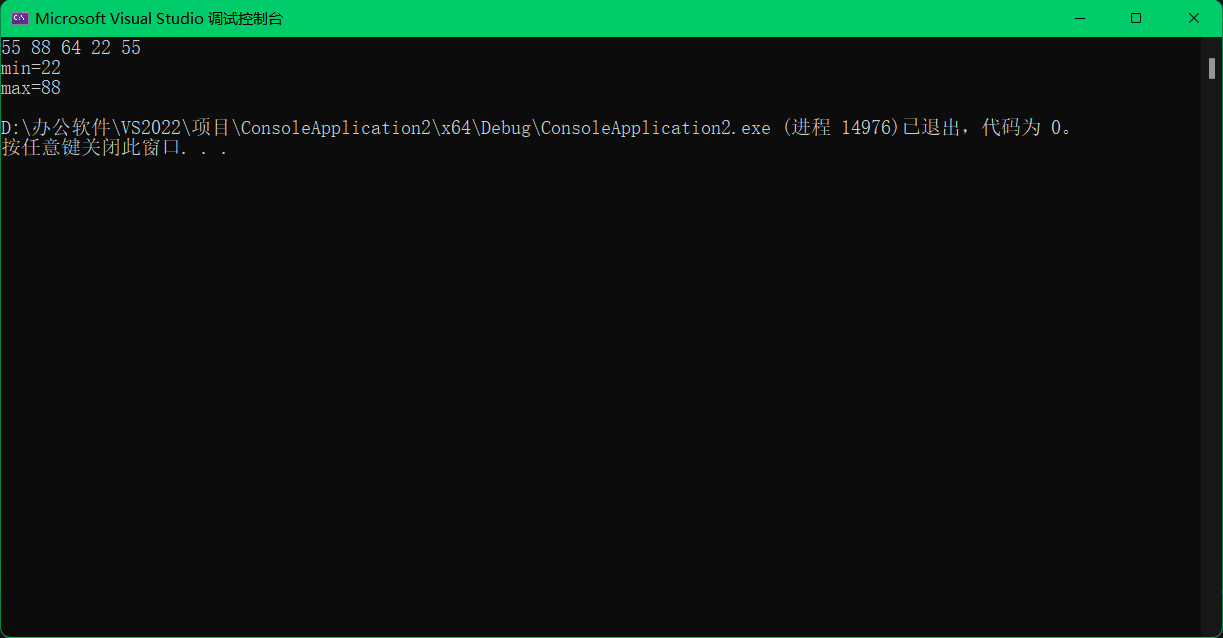
a = x;

}

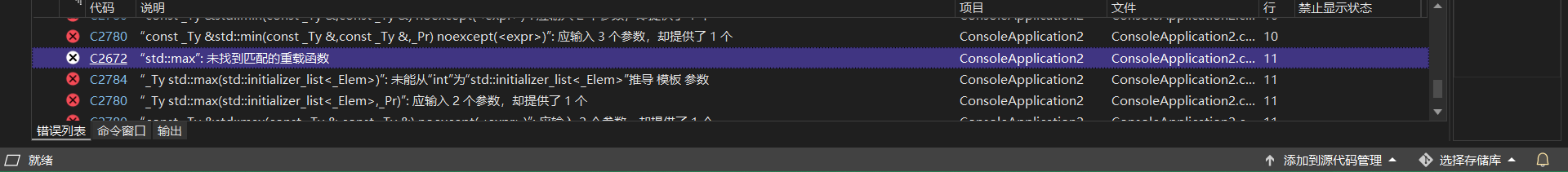
return a;

}

### 程序结果



### 感想心得



子程序中的函数未在主程序中声明，导致程序发生错误。

将单文件程序改为多文件程序，有以下的优点：

1. 便于复用代码。通用性强的重复的功能只要写一遍就可以了，下次要用在其它程序上时只要更改很小的部分或者可以不用更改。  
   2.便于多人协作。在设计软件之初就可以很清楚地分配各个开发部门的任务。模块的编写者本身只要关注他所写的东西，清楚这一部分的功能，留出接口就可以了。另外，对于整个工程的负责人而言，这样会方便浏览全局的工作进度，统筹人员安排。  
   3.便于修改和维护。如果能确定只是某个模块有问题，在模块内解决即可，不需要牵一发而动全身。要升级某一部分的功能，可以只针对具体的模块重新开发，节约成本。

Copyright ©2021-2099 qiuhaolong. All rights reserved