

PTA 系统常见问题解答（FAQ）

评分

试题的解答提交后由评分系统评出即时得分，每一次提交会判决结果会及时通知；系统可能的反馈信息包括：

结 果	说 明
提交成功	对于判断、选择、填空题，系统已经接收到您的提交
稍后显示	对于判断、选择、填空题，在题目集关闭之前，不显示判题结果
已被覆盖	对于判断、选择、填空题，该提交已经被您的当前提交所覆盖，系统将只评判题目集关闭前对该题目的最后一次提交
等待评测	评测系统还没有评测到这个提交，请稍候
正在评测	评测系统正在评测，稍候会有结果
编译错误	您提交的代码无法完成编译，点击“编译错误”可以看到编译器输出的错误信息
答案正确	恭喜！您通过了这道题
部分正确	您的代码只通过了部分测试点，继续努力！
格式错误	您的程序输出的格式不符合要求（比如空格和换行与要求不一致）
答案错误	您的程序未能对评测系统的数据返回正确的结果
运行超时	您的程序未能在规定时间内运行结束
内存超限	您的程序使用了超过限制的内存
异常退出	您的程序在运行过程中崩溃了
非零返回	您的程序结束时返回值非 0，如果使用 C 或 C++ 语言要保证 <code>int main</code> 函数最终 <code>return 0</code>
段错误	您的程序发生段错误，可能是数组越界，堆栈溢出（比如，递归调用层数太多）等情况引起
浮点错误	您的程序运行时发生浮点错误，比如遇到了除以 0 的情况
输出超限	您的程序输出了过多内容，一般可能是无限循环输出导致的结果
内部错误	评测系统发生内部错误，无法评测。工作人员会努力排查此种错误

常见问题

***我应该从哪里读输入，另外应该输出到哪里？**

如果没有特别说明，你的程序应该从标准输入（**stdin**，传统意义上的“**键盘**”）读入，并输出到标准输出（**stdout**，传统意义上的“**屏幕**”），**不要使用文件做输入输出**。由于系

统是在你的程序运行结束后开始检查输出是否是正确的，对于有多组测试数据的输入，可以全部读入之后再输出，也可以处理一组测试数据就输出一组。

***为什么提交后弹出的窗口一直显示“等待评测”？**

提交后即刻弹出的窗口不会自动更新状态。如果看到“等待评测”，请关闭该窗口，去“提交列表”中找到自己的提交，点击分数链接，即可查看详细结果。

***为什么我的程序交在这里得到编译错误，而我在自己的机器上已经编译通过了？**

本系统所使用的编译器和你在自己机器上使用的可能有区别，请留意几个常见的地方：

- 本系统是 64 位 Linux 系统，使用的编译器版本和编译参数可以参见编译器帮助
- Java 代码需使用 Main 作为主类名
- Visual C++ 6.0 和 Turbo C++ 3.0（及它们的更低版本）有较多违背 C++ 标准（ISO/IEC 14882）的地方，不要使用它们来判断 C++ 程序语法上是否有问题
- C++ 下 64 位整数的类型是 long long，不要使用 __int64

***为什么我的程序得到了“非零返回”？**

返回零表示一个程序正常结束，如果没有返回零，则系统认为程序没有正常结束，这时即便输出了正确的内容也不予通过。

- C 或 C++ 代码请确认 int main 函数最终会返回 0，不要声明为 double main 或者 void main
- 有异常的语言，请确认程序处理了可能抛出的异常

***程序的时间和内存占用是如何计算的？**

程序的运行时间为程序在所有 CPU 核占用的时间之和,内存占用取程序运行开始到结束占用内存的最大值。

***为什么同样的程序运行时间和所用内存会不同?**

程序运行时间会受到许多因素的影响,尤其是在现代多任务操作系统以及在使用动态库的情况下,多次使用同一输入运行同一程序所需时间和内存有一些不同是正常现象。我们的题目给出的运行限制一般为标准程序的若干倍,也就是说,选用正确的算法和合适的语言,那么运行限制是富余的。

***不同语言的时间限制和内存限制是相同的吗?**

是相同的,我们认为选择合适的编程语言也是一项必备技能,所以没有为不同语言设置不同的限制条件。

***我提交的代码可以做什么,有什么限制吗?**

没有。这里没有系统调用白名单,也没有针对语言限制可使用的包或库。虽然我们比较宽容大度,但还是请不要做不符合道义的事情。如果你需要使用我们系统没有提供的某个语言的某个库,或者需要更改编译参数,可以联系我们。

其他问题

在考试或比赛中遇到其他问题请咨询现场工作人员。

常见 C 语言程序问题

以下内容摘自浙大版《C 语言程序设计 (第 3 版)》。

(1) `main` 的问题

错误的例子：

```
void main()
{
    printf("hello\n");
}
```

函数 `main()` 的返回类型必须是 `int`，在 `main()` 里一定要有一句

```
return 0;
```

来返回 0 出去。

很多旧的基于 Windows 的 C 编译器来写的书还在用 `void main()`，这是无法接受的。`main()` 的返回值是有意义的，如果返回的不是 0，就表示程序运行过程中错误了，那么服务器上的判题程序也会给出错误的结论。

另外，某些 IDE 需要在 `main()` 的最后加上一句：

```
system("pause");
```

或

```
getch();
```

来形成暂停。在上传代码到 PTA 的时候一定要把这句删了，要不然会得到一个超时错误。

(2) 多余的输出问题

错误的例子：

```
int main()
{
    int a, b;
    printf("请输入两个整数: ");
    scanf("%d %d", &a, &b);
    ...
    printf("%d 和 %d 的最大公约数是 %d\n", a, b, c);
    return 0;
```

```
}
```

程序中不要有任何用户友好性的提示等的输出,只能严格按照题目中所规定的输出格式的要求来输出。

你可以运行自己的程序,用题目中的输入样例来输入,如果得到的输出和输出样例完全相同,一个字符也不多,一个字符也不少,那么这样的格式就是对的。

(3) 汉字问题

程序中不要出现任何汉字,即使在注释中也不行。服务器上使用的文字编码未必和你的电脑上的相同,你认为无害的汉字会被编译器认为是各种奇怪的东西。

(4) 输出格式问题

仔细阅读题目中对于输出格式的要求。因为在服务器上程序是严格地按照预设的输出来比对你的程序的输出。

常见的输出格式问题包括:

- 行末要求不带空格(或带空格)
- 输出要求分行(或不分行)
- 有空格没空格要看仔细
- 输出中的标点符号要看清楚,尤其是绝对不能用中文全角的标点符号,另外单引号 `'` 和一撇 ``` 要分清楚
- 当输出浮点数时,通常题目中会做适当处理,要求比较明确的输出格式,一定要严格遵守,因为浮点数会涉及到输出的精度问题
- 当输出浮点数时,如果可能输出 `0`,而数据可能为负时,有可能出现输出 `-0.0` 的情况,需要自己写代码判断,保证不出现 `-0.0`

(5) 不能用的库函数

某些库函数因为存在安全隐患是不能用的，目前主要常见的是 `itoa` 和 `gets`。

(6) 过时的写法问题

某些旧书上的过时写法也会在服务器的 gcc 编译时引起错误，例如：

```
int f( )  
int a;  
{  
}
```