

The 2015 ACM-ICPC China Shaanxi Provincial
Programming Contest
热身赛



西北工业大学
NORTHWESTERN POLYTECHNICAL UNIVERSITY



acm International Collegiate
Programming Contest

hosted by Northwestern Polytechnical University
Xian, China
June 6, 2015

This problem set should contain two(2) problems on six (6) pages. Please inform a runner immediately if something is missing from your problem set.



问题 A

测试输入

时间限制：3000ms，内存限制：64MB

这个例子用来测试 ACM-ICPC（ACM 国际大学生程序设计竞赛）的输入。在 ACM-ICPC 中，经常有这样几种类型的输入要求：

- ①有很多组数据，组数为 T，每组数据有若干数据；
- ②有很多组数据，每组数据有若干数据，输入直到没有数据为止。
- ③有很多组数据，每组数据有若干数据，输入直到某个条件的数据出现为止。

①的解法可以用下面这个模板

| C 语言 | C++ |
|---|---|
| <pre>#include <stdio.h> int main() { int T; scanf("%d",T); //组数 while (T--) { //...处理每一组 } return 0; }</pre> | <pre>#include <iostream> using namespace std; int main() { int T; cin>>T; //组数 while (T--) { //...处理每一组 } return 0; }</pre> |

②的解法需要利用 scanf 函数的返回值或者 cin 错误标记。当没有输入数据时，scanf 函数返回 EOF，cin>>输入操作符返回流对象的引用的返回值，若流被标记错误（读取失败）就返回 false。

如果这样写程序，调试时需要在最后输入 Ctrl+Z，它代表 EOF。

可以用下面这个模板



| C 语言 | C++ |
|--|--|
| <pre>#include <stdio.h> int main() { int m,n; while (scanf("%d%d",&m,&n)!=EOF) { //...处理每一组 } return 0; }</pre> | <pre>#include <iostream> using namespace std; int main() { int m,n; while (cin>>m>>n) { //...处理每一组 } return 0; }</pre> |

③的解法结合②可以用下面这个模板

| C 语言 | C++ |
|---|--|
| <pre>#include <stdio.h> int main() { int m,n; while (scanf("%d%d",&m,&n)!=EOF && m!=-1) { //...处理每一组 } return 0; }</pre> | <pre>#include <iostream> using namespace std; int main() { int m,n; while (cin>>m>>n && m!=-1) { //...处理每一组 } return 0; }</pre> |



输入

测试数据有多组，每行一组，输入 a 和 b 两个正整数，直到没有数据或者 a 为-1，b 为-1 时。

输出

对于每个测试用例数据，计算 a 和 b 的平均值。

| 输入样例 | 输出样例 |
|-------|------|
| 1 3 | 2 |
| 5 6 | 5 |
| 7 9 | 8 |
| 10 17 | 13 |



问题 B

从文件中输入数据

时间限制：8000ms，内存限制：64MB

ACM-ICPC 竞赛中，反复调试一个程序是常有的事。有时，一道试题有大量输入数据要输入，那么反复输入数据也是频繁的。为了加快输入（ACM 队员不会弱智到一个一个地输吧），有下面这些办法：

- ①把批量的输入数据放到一个文本文件中，随时拿来用；
- ②从文本文件中复制数据，输入时“粘贴”即可；
- ③更好的办法是输入数据来自文本文件，可以用下面这个模板

| C 语言 | C++ |
|--|--|
| <pre>#include <stdio.h> int main() { int T; //输入来自文件 b.in freopen("b.in","r",stdin); //输出到文件 b.out freopen("b.out","w",stdout); scanf("%d",T); //组数 while (T--) { //...处理每一组 } fclose(stdin); fclose(stdout); return 0; }</pre> | <pre>#include <iostream> #include <cstdio> using namespace std; int main() { int T; //输入来自文件 b.in freopen("b.in","r",stdin); //输出到文件 b.out freopen("b.out","w",stdout); cin>>T; //组数 while (T--) { //...处理每一组 } fclose(stdin); fclose(stdout); }</pre> |



| | |
|--|------------------------|
| | <pre>return 0; }</pre> |
|--|------------------------|

`freopen("b.in", "r", stdin);` 函数调用的意思是把标准输入（`scanf`, `cin`）重定向为来自文件 `"b.in"`，而不是键盘。

`freopen("b.out", "w", stdout);` 函数调用的意思是把标准输出（`printf`, `cout`）重定向为到文件 `"b.out"`，而不是显示器。

C++的同学注意，`freopen` 函数需要包含头文件：`#include <cstdio>`。

当然，`freopen` 函数是为了调试，**在提交评阅时，记得将它们删除（注释掉）**。

输入

测试数据有 `T` 组，每组数据第 1 行为数组长度，第 2 行为数组每个数据，数据均是整型，数组元素最多 100 个。

输出

对每组数组由小到大排序后输出，数据间用空格隔开，每组一行。

| 输入样例 | 输出样例 |
|----------------------|----------------------|
| 2 | 1 2 7 8 9 |
| 5 | 1 2 3 4 5 6 7 8 9 10 |
| 7 8 9 1 2 | |
| 10 | |
| 10 9 8 7 6 5 4 3 2 1 | |