

ACM程序设计

输入输出技术





ACM题目特点：

由于ACM竞赛题目的输入数据和输出数据一般有多组（不定），并且格式多种多样，所以，如何处理题目的输入输出是对大家的一项最基本的要求。这也是困扰初学者的一大问题。

下面，分类介绍：



先看一个超级简单的题目：

- <http://acm.hdu.edu.cn/showproblem.php?pid=1089>
- Sample input:
 - 1 5
 - 10 20
- Sample output:
 - 6
 - 30



初学者很常见的一种写法:

- `#include<stdio.h>`
- `void main()`
- `{`
- `int a,b;`
- `scanf(“%d %d”,&a,&b);`
- `Printf(“%d”,a+b);`
- `}`



有什么问题呢？

这就是下面需要解决的问题





第二部分

基本输入输出



输入_第一类:

- 输入不说明有多少个Input Block,以EOF为结束标志。
参见: HDOJ_1089
- <http://acm.hdu.edu.cn/showproblem.php?pid=1089>



Hdoj_1089源代码:

```
#include <stdio.h>
int main()
{
    int a,b;
    while(scanf("%d %d",&a, &b) != EOF)
        printf("%d\n",a+b);
}
```




本类输入解决方案:

- C语法:

```
while(scanf("%d %d",&a, &b) != EOF)
{
    ....
}
```

- C++语法:

```
while( cin >> a >> b )
{
    ....
}
```



说明（1）：

1. **Scanf**函数返回值就是读出的变量个数，
如：`scanf("%d %d", &a, &b);`
如果只有一个整数输入，返回值是**1**，
如果有两个整数输入，返回值是**2**，如
果一个都没有，则返回值是**-1**。
2. **EOF**是一个预定义的常量，等于**-1**。



输入_第二类:

- 输入一开始就会说有N个Input Block,下面接着是N个Input Block。
参见: HDOJ_1090
- <http://acm.hdu.edu.cn/showproblem.php?pid=1090>



Hdoj_1090源代码:

```
#include <stdio.h>
int main()
{
    int n,i,a,b;
    scanf("%d",&n);
    for(i=0;i<n;i++)
    {
        scanf("%d %d",&a, &b);
        printf("%d\n",a+b);
    }
}
```

2012-7-11



本类输入解决方案:

- C语法:

```
scanf("%d",&n) ;  
for( i=0 ; i<n ; i++ )  
{  
    ....  
}
```

- C++语法:

```
cin >> n;  
for( i=0 ; i<n ; i++ )  
{  
    ....  
}
```

2012-7-11



输入_第三类:

- 输入不说明有多少个Input Block,但以某个特殊输入为结束标志。

参见: HDOJ_1091

- <http://acm.hdu.edu.cn/showproblem.php?pid=1091>



Hdoj_1091源代码:

```
#include <stdio.h>
int main()
{
    int a,b;
    while(scanf("%d %d",&a, &b) &&(a!=0 && b!=0))
        printf("%d\n",a+b);
}
```

上面的程序有什么问题？



本类输入解决方案:

- C语法:

```
while(scanf("%d",&n) && n!=0 )  
{  
    ....  
}
```

- C++语法:

```
while( cin >> n && n != 0 )  
{  
    ....  
}
```




输入_第四类:

- 以上几种情况的组合
- <http://acm.hdu.edu.cn/showproblem.php?pid=1092>
- <http://acm.hdu.edu.cn/showproblem.php?pid=1093>
- <http://acm.hdu.edu.cn/showproblem.php?pid=1094>



输入_第五类:

- 输入是一整行的字符串的
参见: HDOJ_1048
- <http://acm.hdu.edu.cn/showproblem.php?pid=1048>



本类输入解决方案:

- C语法:

```
char buf[20];  
gets(buf);
```

- C++语法:

如果用string buf;来保存:

```
getline( cin , buf );
```

如果用char buf[255]; 来保存:

```
cin.getline( buf, 255 );
```




说明（5_1）：

- `scanf(“ %s%s”,str1,str2)`，在多个字符串之间用一个或多个空格分隔；
- 若使用`gets`函数，应为`gets(str1); gets(str2);`字符串之间用回车符作分隔。
- 通常情况下，接受短字符用`scanf`函数，接受长字符用`gets`函数。
- 而`getchar`函数每次只接受一个字符，经常`c=getchar()`这样来使用。



说明 (5_2) : `cin.getline` 的用法:

- `getline` 是一个函数，它可以接受用户的输入的字符，直到已达指定个数，或者用户输入了特定的字符。它的函数声明形式（函数原型）如下：
`istream& getline(char line[], int size, char endchar = '\n');`
- 不用管它的返回类型，来关心它的三个参数：
- `char line[]`: 就是一个字符数组，用户输入的内容将存入在该数组内。
- `int size` : 最多接受几个字符？用户超过`size`的输入都将不被接受。
- `char endchar` : 当用户输入`endchar`指定的字符时，自动结束。默认是回车符。



说明 (5_2) 续

- 结合后两个参数，`getline`可以方便地实现：用户最多输入指定个数的字符，如果超过，则仅指定个数的前面字符有效，如果没有超过，则用户可以通过回车来结束输入。
- `char name[4];`
- `cin.getline(name,4,'\n');`
- 由于 `endchar` 默认已经是 `'\n'`，所以后面那行也可以写成：
- `cin.getline(name,4);`



思考:

- 以下题目属于哪一类输入?
- <http://acm.hdu.edu.cn/showproblem.php?pid=1018>
- <http://acm.hdu.edu.cn/showproblem.php?pid=1013>



输出_第一类:

- 一个Input Block对应一个Output Block, Output Block之间没有空行。
参见: HDOJ_1089
- <http://acm.hzjee.edu.cn/showproblem.php?pid=1089>



解决方案:

- C语法:

```
{  
    ....  
    printf("%d\n",ans);  
}
```

- C++语法:

```
{  
    ...  
    cout << ans << endl;  
}
```



输出_第二类:

- 一个Input Block对应一个Output Block, 每个Output Block之后都有空行。
参见: HDOJ_1095
- <http://acm.hdu.edu.cn/showproblem.php?pid=1095>



1095源代码

```
#include <stdio.h>
int main()
{
    int a,b;
    while(scanf("%d %d",&a, &b) != EOF)
        printf("%d\n\n",a+b);
}
```




解决办法:

- C语法:

```
{  
    ....  
    printf("%d\n\n",ans);  
}
```

- C++语法:

```
{  
    ...  
    cout << ans << endl << endl;  
}
```



输出_第三类:

- 一个Input Block对应一个Output Block, Output Block之间有空行。
参见: HDOJ_1096
- <http://acm.hdu.edu.cn/showproblem.php?pid=1096>



1096源代码

```
■ #include <stdio.h>
■ int main()
■ {
■     int icalse,n,i,j,a,sum;
■     scanf("%d",&icalse);
■     for(i=0;i<icalse;i++)
■     {
■         sum=0;
■         scanf("%d",&n);
■         for(j=0;j<n;j++)
■         {
■             scanf("%d",&a);
■             sum+=a;
■         }
■         if(i<icalse-1)
■             printf("%d\n\n",sum);
■         else
■             printf("%d\n",sum);
■     }
■ }
```

2012-7-11



解决办法:

- C语法:

```
for (k=0;k<count;k++)  
{  
    while (...)  
    {  
        printf(" %d\n",result);  
    }  
    if (k!=count-1) printf("\n");  
}
```

- C++语法:

类似，输出语句换一下即可。



思考:

- 以下题目属于哪一类输出?
- <http://acm.hdu.edu.cn/showproblem.php?pid=1016>
- <http://acm.hdu.edu.cn/showproblem.php?pid=1017>



附:

初学者常见问题



一、编译错误

- Main函数必须返回int类型(正式比赛)
- 不要在for语句中定义类型
- __int64不支持，可以用long long代替
- 使用了汉语的标点符号
- itoa不是ansi函数
 - 能将整数转换为字符串而且与ANSI标准兼容的方法是使用sprintf()函数

```
int num = 100;
char str[25];
sprintf(str, "%d", num);
```
- 另外，拷贝程序容易产生错误



下面的hdoj1089为什么 CE?

```
#include <stdio.h>
int main()
{
    int a,b;
    while(scanf("%d %d",&a, &b) != EOF)
        printf("%d\n",a+b);
}
```



二、小技巧

- 数据的拷贝（特别是输出的提示信息）
- 调试的sample input的拷贝



三、C语言处理“混合数据”的问题

<http://acm.hdu.edu.cn/showproblem.php?pid=1170>



常见的代码:

-
- **scanf("%d\n",&icase);**
- **for (i=0;i<icase;i++)**
- **{**
- **scanf("%c%d%d",&opera,&num1,&num2);**
-
- **}**
-



上面程序
有什么问题？



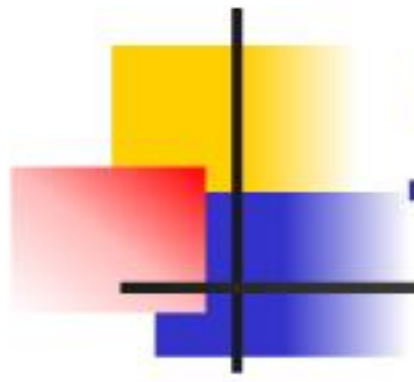
四、Printf和cout混用的问题

- 以下的程序输出什么？
- #include<stdio.h>
- #include<iostream.h>
- int main()
- {
- int j=0;
- for(j=0;j<5;j++)
- {
- cout<<"j=";
- printf("%d\n",j);
- }
- return 0;
- }



为什么？

- 一个带缓冲输出 (cout)
- 一个不带缓冲输出 (printf)



五、输入输出原理

Input

1 5

2 6

10 20

111 111

321 123

Output

6

8

30

222

444



初步体会ACM的魅力:

Given two non-negative integers m and n , you will have to find the last digit of m^n in decimal number system.

- **Input**
- The input file contains several lines. Each line contains two integers m and n (both less than 10^{1001}). Input is terminated by a line containing two zeroes. This line should not be processed.
- **Output**
- For each set of input you must produce one line of output which contains a single digit. This digit is the last digit of m^n .
- **Sample Input**
- 3 2
- 3 5
- 0 0
- **Sample Output**
- 9
- 3



学习方式

- 练习->总结->练习->总结->.....
- <http://acm.hdu.edu.cn>
- 各种ACM论坛
- google、baidu



杭州电子科技大学



HANGZHOU DIANZI UNIVERSITY

Online Judge



Online Judge	Problem Set	Authors	Online Contests	Exercise Author
Notification F.A.Q Forum Blog Administration	Problem Archive Submit Solution Realtime Judge Status <input type="text"/> <input type="button" value="Search"/>	Register New Author Update Your Information Authors Ranklist <input type="text"/> <input type="button" value="Search"/>	Next Contest Time 2007-10-13 18:00:00 (GMT+8) Scheduled Contests Passed Contests DIY Contests !	Author ID <input type="text"/> Password <input type="text"/> <input type="button" value="Sign In"/> <input type="button" value="Register new ID"/>

Welcome to HDU Online Judge System

Problem Set is the place where you can find large amount of problems from different programming contests.

First of all, read carefully [Frequently Asked Questions](#).

Then, choose [Problem](#), solve it and submit it. To submit your solution, please, use [This Page](#).

If you want to publish your problems or setup your own online contest, just [Contact Us](#).

If you have some questions, just discuss on our [Forum](#)!

We provide some softwares and documents, to download them, [Just Click Here](#).

- 1、系统新增"DIY Contests"功能，欢迎测试，有问题请到[这里](#)跟帖，谢谢
- 2、“HDU 2007'10程序设计竞赛”将于10月20号左右进行，有志参赛的同学可以提前准备了~
- 3、如何在HDU添加自己的题目或者举办自己的比赛？



课后任务：

1、熟悉<http://acm.hdu.edu.cn>

2、完成在线练习（DIY-2111）：

《ACM程序设计》在线作业（1）

——基本输入输出练习

3、学有余力，可以尝试下面题目：

1016-1018、1013、1061

1170、2000-2043