



链滴

# PAT 甲级刷题实录——1001

作者: [aopstudio](#)

原文链接: <https://ld246.com/article/1578200348944>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

## 前言

先说个挺不好意思的事，小弟我脑子一热，考研报了浙大。更不好意思的是，自己竟然还报有着能进试的想法。浙大复试需要上机考试，基本上考试内容和PAT考试差不多，所以现在就得开始刷题准备也不管能不能进复试吧，就算没进复试，去找工作的话练练算法总是有好处的，大学前三年确实没有么太注重算法和编码的练习，现在也该补一补。

因为我前三年半没怎么刷过算法题，所以我的基础是很差的。也不怕大家笑话，我就从最简单的题开一道一道练习。那么今天练习的就是甲级第一题。

## 原题

Calculate  $a+b$  and output the sum in standard format -- that is, the digits must be separated nto groups of three by commas (unless there are less than four digits).

### Input Specification:

Each input file contains one test case. Each case contains a pair of integers  $a$  and  $b$  where  $-10^6 \leq a, b \leq 10^6$ . The numbers are separated by a space.

### Output Specification:

For each test case, you should output the sum of  $a$  and  $b$  in one line. The sum must be written in the standard format.

### Sample Input:

```
-1000000 9
```

### Sample Output:

```
-999,991
```

## 解析

这题乍一看很简单，就是输出两个数的和，但是有个比较复杂的地方，就是每隔千位得加上逗号，变西方打印的数字形式。说句题外话，这种写法完全就只是为了西方人的方便而考虑的，因为他们是以为基本单位来读较大的数的，比如thousand（千），million（百万），billion（十亿）。而中国人则是万为基本单位来读较大的数的，比如万，亿，兆，这样的写法反而会带来不方便。但没办法，题目里求这么做，那就只能这么做了。

## 解法

PAT考试理论上是可以各种语言写程序的（抱着我的Java摩拳擦掌中）。不过我刚刚看了一下，用语言和C++语言之外的其他语言基本上就是作死，因为速度太慢了（啊，我最爱的Java）。甚至还有面这种恐怖的说法：



陈越姥姥 (作者) 回复 华阿炆

3 年前

所以另加100毫秒就是给Java启动用的，所以用Java就不要用Scanner，否则C加的蜗牛都绕地球两圈了，Java家连输入都没读完，这个实在没办法照顾了。

但我还是不死心，想看看用Java写到底是怎么样的效果，之后再用C语言或者C++重写一遍对比一下。

这道题我的思路是这样的：**把相加的结果转换为字符数组，之后逐个输出数字，当数组长度和输出下的差减1能够整除3的时候，就输出逗号。**

Java代码如下：

```
import java.io.*;
public class Main {

    public static void main(String[] args) throws IOException {
        BufferedReader br=new BufferedReader(new InputStreamReader(System.in));
        String st=br.readLine();
        String[] nums=st.split(" ");
        int a=Integer.parseInt(nums[0]);
        int b=Integer.parseInt(nums[1]);
        int c=a+b;
        String str=String.valueOf(c);
        int n=str.length();
        for(int i=0;i<n;i++) {
            System.out.print(str.charAt(i));
            if(c<0&&i!=n-1&&(n-i-1)%3==0&&i!=0) {
                System.out.print(',');
            }
            if(c>0&&(n-i-1)%3==0&&i!=n-1) {
                System.out.print(',');
            }
        }
    }
}
```

因为老师说别用Scanner，所以我都不敢用Scanner而是换成了BufferedReader。应该说Java要是不Scanner，输入特别麻烦，因为BufferedReader一次只能读一行，所以我在读完一行之后还得再用split方法把参数分割开。特别要注意的是当两数之和为负数的情况，此时若 $i=0$ ，则不能输出逗号，否则输出-999,999这样的数字。

这段程序是能够通过测试的，执行情况如图所示：

提交时间	状态	分数	题目	编译器	耗时	用户
2020/1/5 12:43:25	答案正确	20	1001	Java (openjdk)	109 ms	程艺工作室
测试点	结果		耗时		内存	
0	答案正确		95 ms		10204 KB	
1	答案正确		109 ms		10220 KB	
2	答案正确		92 ms		10220 KB	
3	答案正确		92 ms		10280 KB	
4	答案正确		98 ms		10184 KB	
5	答案正确		91 ms		10160 KB	
6	答案正确		92 ms		10016 KB	
7	答案正确		94 ms		10224 KB	
8	答案正确		94 ms		9956 KB	
9	答案正确		94 ms		9928 KB	
10	答案正确		103 ms		10196 KB	
11	答案正确		93 ms		10124 KB	

因为我没什么做在线算法题的经验，也不知道这个运行效率算什么水平（后来才发现完全不行）。又尝试用C++写了一遍，代码如下：

```
#include <iostream>
#include <string>
using namespace std;

int main()
{
    int a,b,c,n;
    cin>>a>>b;
    c=a+b;
    string str=to_string(c);
    n=str.size();
    for(int i=0;i<n;i++){
        cout<<str[i];
        if(c<0&&!i=n-1&&(n-i-1)%3==0&&!i=0) {
            cout<<' ';
        }
        if(c>0&&(n-i-1)%3==0&&!i=n-1) {
            cout<<' ';
        }
    }
}
```

```
}  
return 0;  
}
```

这段代码的运行结果如图：

提交时间	状态	分数	题目	编译器	耗时	用户
2020/1/5 12:30:23	答案正确	20	1001	C++ (g++)	5 ms	程艺工作室
测试点	结果	耗时	内存			
0	答案正确	3 ms	512 KB			
1	答案正确	4 ms	376 KB			
2	答案正确	5 ms	380 KB			
3	答案正确	3 ms	380 KB			
4	答案正确	4 ms	384 KB			
5	答案正确	3 ms	384 KB			
6	答案正确	3 ms	384 KB			
7	答案正确	3 ms	376 KB			
8	答案正确	3 ms	384 KB			
9	答案正确	3 ms	356 KB			
10	答案正确	4 ms	384 KB			
11	答案正确	3 ms	384 KB			

可以看到效率和Java相比完全就是一个天上，一个地下，差距也太大了吧。看来以后还是得用C++来题。

## 歪门邪道

网上看到一个自己啥也不用干，只要调用Java包装好的方法的做法，也把代码贴在这里：

```
import java.text.DecimalFormat;  
import java.util.Scanner;  
  
public class Main {  
    public static void main(String[] args) throws IOException {  
        Scanner in = new Scanner(System.in);  
        DecimalFormat df = new DecimalFormat("#,###");  
        while (in.hasNextInt()) {  
            int a = in.nextInt();  
            int b = in.nextInt();  
        }  
    }  
}
```

```
System.out.println(df.format(a+b));  
    }  
}  
}
```

这个运行效率就更低了，如图：

提交时间	状态	分数	题目	编译器	耗时	用户
2020/1/5 12:55:55	答案正确	20	1001	Java (openjdk)	129 ms	程艺工作室
测试点	结果		耗时		内存	
0	答案正确		123 ms		11420 KB	
1	答案正确		122 ms		11392 KB	
2	答案正确		121 ms		11320 KB	
3	答案正确		123 ms		11652 KB	
4	答案正确		115 ms		11268 KB	
5	答案正确		120 ms		11424 KB	
6	答案正确		118 ms		11560 KB	
7	答案正确		129 ms		11616 KB	
8	答案正确		118 ms		11384 KB	
9	答案正确		118 ms		11620 KB	
10	答案正确		118 ms		11584 KB	
11	答案正确		125 ms		11376 KB	