问题描述

求1+2+3+...+n的值。

输入格式

输入包括一个整数n。

输出格式

输出一行，包括一个整数，表示1+2+3+...+n的值。

样例输入

4

样例输出

10

样例输入

100

说明：有一些试题会给出多组样例输入输出以帮助你更好的做题。

一般在提交之前所有这些样例都需要测试通过才行，但这不代表这几组样例数据都正确了你的程序就是完全正确的，潜在的错误可能仍然导致你的得分较低。

样例输出

5050

数据规模与约定

1 <= n <= 1,000,000,000。

说明：请注意这里的数据规模。

本题直接的想法是直接使用一个循环来累加，然而，当数据规模很大时，这种“暴力”的方法往往会导致超时。此时你需要想想其他方法。你可以试一试，如果使用1000000000作为你的程序的输入，你的程序是不是能在规定的上面规定的时限内运行出来。

本题另一个要值得注意的地方是答案的大小不在你的语言默认的整型(int)范围内，如果使用整型来保存结果，会导致结果错误。

如果你使用C++或C语言而且准备使用printf输出结果，则你的格式字符串应该写成%I64d以输出long long类型的整数。

锦囊1

使用求和公式计算结果，使用64位整型(C++:long long, C:long long, Java:long)来保存结果和进行中间运算。

锦囊2

利用公式1+2+3+...+n=(1+n)\*n/2来计算。注意这里的乘法运算时需要在64位整数下进行。

C代码

#include <stdio.h>

int main()

{

int n;

long long ret = 0, tmp;

scanf("%d", &n);

tmp = n;

printf("%I64d", (1+tmp) \* tmp / 2);

return 0;

}

C++代码

#include <iostream>

using namespace std;

int main()

{

long long n;

cin >> n;

cout << (1+n) \* n / 2;

return 0;

}

Java代码

import java.io.BufferedReader;

import java.io.IOException;

import java.io.InputStreamReader;

public class Main {

public static void main(String args[]) throws NumberFormatException, IOException{

BufferedReader strin = new BufferedReader(new InputStreamReader(System.in));

long i = Integer.parseInt(strin.readLine());

long sum = (1+i)\*i/2;

System.out.println(sum);

}

}