

P1464 Function

题目描述

对于一个递归函数 $w(a,b,c)$

如果 $a \leq 0$ or $b \leq 0$ or $c \leq 0$ 就返回值1.

如果 $a > 20$ or $b > 20$ or $c > 20$ 就返回 $w(20,20,20)$

如果 $a < b$ 并且 $b < c$ 就返回 $w(a,b,c-1) + w(a,b-1,c-1) - w(a,b-1,c)$

其它的情况就返回 $w(a-1,b,c) + w(a-1,b-1,c) + w(a-1,b,c-1) - w(a-1,b-1,c-1)$

这是个简单的递归函数，但实现起来可能会有些问题。当 a,b,c 均为15时，调用的次数将非常的多。你要想个办法才行。

absi2011：比如 $w(30,-1,0)$ 既满足条件1又满足条件2

这种时候我们就按最上面的条件来算

所以答案为1

输入格式

会有若干行。

并以-1,-1,-1结束。

保证输入的数在 $[-9223372036854775808, 9223372036854775807]$ 之间，并且是整数。

输出格式

输出若干行，每一行格式：

$w(a, b, c) = ans$

注意空格。

输入样例

```
1 1 1
2 2 2
-1 -1 -1
```

输出样例

```
w(1, 1, 1) = 2
w(2, 2, 2) = 4
```

解析

简单的模拟，注意需要加上缓存。

编码

```
#include <bits/stdc++.h>

using namespace std;

long long f[25][25][25];
long long a, b, c;

long long Cout(long long a, long long b, long long c) {
    //条件一、
    if (a <= 0 || b <= 0 || c <= 0) {
        return 1;
    }
    //条件二、
    if (a > 20 || b > 20 || c > 20) {
        return Cout(20, 20, 20);
    }
    //检查缓存
    //放在这里是防止数据过大导致数组溢出
    if (f[a][b][c] != 0) {
        return f[a][b][c];
    }
    //条件三、
    if (a < b && b < c) {
        return f[a][b][c] = Cout(a, b, c - 1) +
            Cout(a, b - 1, c - 1) -
            Cout(a, b - 1, c);
    }
    //条件四、
    else {
        return f[a][b][c] =
            Cout(a - 1, b, c) + Cout(a - 1, b - 1, c) +
            Cout(a - 1, b, c - 1) -
            Cout(a - 1, b - 1, c - 1);
    }
}

int main(int argc, char **argv) {
    while (cin >> a >> b >> c) {
        if (a == -1 && b == -1 && c == -1) {
            break;
        }
        int res = Cout(a, b, c);
        printf("w(%d, %d, %d) = %d\n", a, b, c, res);
    }
    return 0;
}
```

}

逻辑航线培优教育，信息学奥赛培训专家。

扫码添加作者获取更多内容。

