

## 1333: 【例2-2】Blah数集

## 题目描述

大数学家高斯小时候偶然间发现一种有趣的自然数集合Blah，对于以 $a$ 为基的集合 $Ba$ 定义如下：

- (1)  $a$ 是集合 $Ba$ 的基，且 $a$ 是 $Ba$ 的第一个元素；
- (2) 如果 $x$ 在集合 $Ba$ 中，则 $2x+1$ 和 $3x+1$ 也都在集合 $Ba$ 中；
- (3) 没有其他元素在集合 $Ba$ 中了。

现在小高斯想知道如果将集合 $Ba$ 中元素按照升序排列，第 $N$ 个元素会是多少？

## 输入格式

输入包括很多行，每行输入包括两个数字，集合的基 $a$  ( $1 \leq a \leq 50$ ) 以及所求元素序号 $n$  ( $1 \leq n \leq 1000000$ )。

## 输出格式

对于每个输入，输出集合 $Ba$ 的第 $n$ 个元素值。

## 输入样例

```
1 100
28 5437
```

## 输出样例

```
418
900585
```

## 解析

单独计算 $2x+1$ 或者 $3x+1$ 是很简单的，麻烦的是我们需要对生成的数字进行排序和去重。因此，我们不能一股脑的将所有数据都进行存储，而应该从所有生成的数据中选择最小的一个进队，如果队列中存在多个相同的数据，则只放入一个。

## 编码

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

```

using namespace std;
int main() {
    int n, m;
    while (cin >> n >> m) {
        //最终计算完毕的队列
        queue<int> q;
        //2倍队列
        queue<int> q1;
        //3倍队列
        queue<int> q2;
        m--;
        //将基准数字存入队列
        q.push(n);
        while (m--) {
            //计算两倍数
            q1.push(q.front() * 2 + 1);
            //计算三倍数
            q2.push(q.front() * 3 + 1);
            //选择两倍数和三倍数中最小的一个存入基础队列
            if (q1.front() > q2.front()) {
                q.push(q2.front());
                q2.pop();
            } else if (q1.front() < q2.front()) {
                q.push(q1.front());
                q1.pop();
            }
            //若相等，则同时放入，别忘了同时删除
            else {
                q.push(q1.front());
                q1.pop();
                q.push(q2.front());
                q2.pop();
            }
            //将已经计算过的基准删除
            q.pop();
        }
        cout << q.front() << endl;
        q.pop();
    }
    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。

