# 1032 The 3n + 1 problem

### 一、题目

#### 问题描述

Problems in Computer Science are often classified as belonging to a certain class of problems (e.g., NP, Unsolvable, Recursive). In this problem you will be analyzing a property of an algorithm whose classification is not known for all possible inputs.

Consider the following algorithm:

1. input n
2. print n
3. if n = 1 then STOP
4. if n is odd then n <- 3n + 1
5. else n <- n / 2
6. GOTO 2

Given the input 22, the following sequence of numbers will be printed 22 11 34 17 52 26 13 40 20 10 5 16 8 4 2 1

It is conjectured that the algorithm above will terminate (when a 1 is printed) for any integral input value. Despite the simplicity of the algorithm, it is unknown whether this conjecture is true. It has been verified, however, for all integers n such that 0 < n < 1,000,000 (and, in fact, for many more numbers than this.)

Given an input n, it is possible to determine the number of numbers printed (including the 1). For a given n this is called the cycle-length of n. In the example above, the cycle length of 22 is 16.

For any two numbers i and j you are to determine the maximum cycle length over all numbers between i and j.

#### 输入数据

The input will consist of a series of pairs of integers i and j, one pair of integers per line. All integers will be less than 1,000,000 and greater than 0.

You should process all pairs of integers and for each pair determine the maximum cycle length over all integers between and including i and j.

You can assume that no opperation overflows a 32-bit integer.

#### 输出数据

For each pair of input integers i and j you should output i, j, and the maximum cycle length for integers between and including i and j. These three numbers should be separated by at least one space with all three numbers on one line and with one line of output for each line of input. The integers i and j must appear in the output in the same order in which they appeared in the input and should be followed by the maximum cycle length (on the same line).

#### 输入样例

1 10  
100 200  
201 210  
900 1000

#### 输出样例

1 10 20  
100 200 125  
201 210 89  
900 1000 174

#### 题目来源

HDU 1032 http://acm.hdu.edu.cn/showproblem.php?pid=1032

### 二、题解

#### 解题思路

在整数i和j之间找到最长的循环次数，输出循环次数。就是一个循环遍历i到j，打擂台地找出最大的循环次数。

#### 参考程序

#include <stdio.h>  
int a[10000];  
long int search(int, int);  
long int check(int);  
int main()  
{  
 int i, j;  
 while(scanf("%d%d", &i, &j)!=EOF)  
 printf("%d %d %ld\n",i,j,search(i, j));  
 return 0;  
}  
long int search(int i, int j)  
{  
 int temp,k,t1,t2;  
 long int a\_max=-1;  
 if (i > j) { temp = j; j = i; i = temp; }  
 for (k = i; k <= j; k++)  
 {  
 t1=check(k);  
 if (t1 > a\_max)a\_max = t1;  
 }  
 return a\_max;  
}  
long int check(int x)  
{  
 long int count = 1;  
 while (x != 1)  
 {  
 if (x % 2 == 0)x = x / 2;  
 else x = 3 \* x + 1;  
 count++;  
 }  
 return count;  
}

#### 复杂度分析

无

#### 编程技巧

无