## 1061 **Rightmost Digit**

### 一、题目

#### 问题描述

Given a positive integer N, you should output the most right digit of N^N.

#### 输入数据

The input contains several test cases. The first line of the input is a single integer T which is the number of test cases. T test cases follow. Each test case contains a single positive integer N(1<=N<=1,000,000,000).

#### 输出数据

For each test case, you should output the rightmost digit of N^N.

#### 输入样例

2

3

4

#### 输出样例

7

6

*Hint*

In the first case, 3 \* 3 \* 3 = 27, so the rightmost digit is 7. In the second case, 4 \* 4 \* 4 \* 4 = 256, so the rightmost digit is 6.

#### 题目来源

HDU 1061 http://acm.hdu.edu.cn/showproblem.php?pid=1061

### 二、题解

#### 解题思路

涉及到快速幂算法，又要用到离散数学和数论的公式。 a^b mod c = ( (a mod c)*（b mod c）) mod c,即a的b次幂对c求余等价于a对c求余乘以b对c求余的结果； 本题求n^n mod 10,等价于(（n mod 10）*（n mod 10）) mod 10; 中间调用递归不断的对n值进行缩小； 当n为奇数时，相对于偶数多乘了一次本身对10的求余。

#### 参考程序

#include <stdio.h>  
int f(int k,int m)  
{  
 int t;  
 if(m==0)  
 return 0;  
 if(m==1)  
 return k;  
 t=f(k,m/2);  
 t\*=t;  
 if(m%2==1)  
 t\*=k;  
 return t%10;  
}  
  
int main()  
{  
 int n;  
 scanf("%d",&n);  
 while(n--)  
 {  
 int m;  
 scanf("%d",&m);  
 printf("%d\n",f(m%10,m));  
 }  
 return 0;  
}

#### 复杂度分析

无

#### 编程技巧

无