**Replace the Bit**

Show Topic Tags   

[Microsoft](http://practice.geeksforgeeks.org/company/Microsoft/)

Given two numbers n and k, change the kth bit of the number n to '0' if it is  '1', else return the number n itself.

**Input:**  
First line of the input contains an intger T denoting the number of test case. Each test contains a single line containg two space seperated integers n and k respectively.

**Output:**  
For each test case output a single integer.

**Constraints:**  
1<=T<=100  
1<=N<=106

**Example:**  
**Input:**  
2  
13 3  
13 2  
**Output:**  
13  
9

**Explanation:**

**Test Case 1:**n = 13 ('1101') k=3  
return 13

**Test Case 2:**n = 13('1101') k=2  
return 9('1001')

\*\*For More Examples Use Expected Output\*\*

<http://practice.geeksforgeeks.org/problems/replace-the-bit/0>

/\*

\* To change this template, choose Tools | Templates

\* and open the template in the editor.

\*/

package javaapplication243;

import java.io.\*;

import java.util.\*;

/\*\*

\*

\* @author Administrador

\*/

public class JavaApplication243 {

/\*\*

\* @param args the command line arguments

\*/

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

// TODO code application logic here

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

int t = Integer.parseInt(br.readLine());

while(t-- > 0) {

String[] input = br.readLine().trim().split(" ");

int n = Integer.parseInt(input[0]);

int k = Integer.parseInt(input[1]);

char[] bin = Integer.toString(n, 2).toCharArray();

bin[k-1] = '0';

String bin2 = new String(bin);

System.out.println(Integer.parseInt(bin2,2));

}

}

}