**Unique Numbers**

Show Topic Tags   

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

In given range, print all numbers having unique digits. e.g. In range 1 to 20 should print all numbers except 11.

**Input:**  
First line consists of T test cases.  
The only line of every test case consists of two integers m and n separated by a space.

**Output:**  
For each test case print all unique numbers separated by space.

**Constraints:**  
1<=T<=100  
1<=m,n<1000

**Example:  
Input:**  
1  
10 20  
**Output:**  
10 12 13 14 15 16 17 18 19 20

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

<http://practice.geeksforgeeks.org/problems/unique-numbers/0>

/\*

\* To change this template, choose Tools | Templates

\* and open the template in the editor.

\*/

package javaapplication248;

import java.io.BufferedReader;

import java.io.IOException;

import java.io.InputStreamReader;

import java.util.ArrayList;

/\*\*

\*

\* @author Administrador

\*/

public class JavaApplication248 {

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

int m = Integer.parseInt(mn[0]);

int n = Integer.parseInt(mn[1]);

for(int i =m; i<=n; i++) {

int[] contDig = new int[10];

int copia = i;

while(copia > 0) {

contDig[copia%10]++;

copia/=10;

}

boolean repets = false;

for(int j =0; j<10; j++) {

if(contDig[j] > 1) {

repets = true;

break;

}

}

if(!repets) {

System.out.print(i + " ");

}

}

System.out.println();

}

}

}