**The Prime Twins**

[maths](http://www.practice.geeksforgeeks.org/tag-page.php?tag=maths&isCmp=0)[sieve](http://www.practice.geeksforgeeks.org/tag-page.php?tag=sieve&isCmp=0)

Given two positive numbers **L** and **R** **(L < R)**, find out the smallest twins in the range **[L , R]**.Two numbers are twins if they are **prime numbers**and there difference is 2.

**Input**  
The first line of input contains an integer **T**denoting the number of test cases. Then **T**test cases follow.   
The first line of each test case contains two positve integers **L** and **R** **(L < R)**, denoting the range **[L < R]**.

**Output**  
Print out the smallest twins in the given range. If no such twins exist, then print **-1**.

**Constraints**  
1 <= **T** <= 200  
0 <= **L** <= 500  
0 <   **R** <= 1000

**Examples**

**Input**  
4  
10 100  
20 50  
850 900  
15 18

**Output**  
11 13  
29 31  
857 859  
-1

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

<http://www.practice.geeksforgeeks.org/problem-page.php?pid=946>

#include <iostream>

#include <stdio.h>

#include <math.h>

using namespace std;

bool esPrimo(int n ) {

    if(n < 2) return false;

    if(n == 2) return true;

    if(n%2==0) return false;

    int sqr = (int) sqrt(n);

    for(int i =3; i<=sqr; i+=2) {

      if(n % i ==0) {

           return false;

        }

    }

    return true;

}

int main() {

    int t;

    scanf("%d",&t);

    while(t--) {

      int l,r;

      scanf("%d %d", &l, &r);

      bool twins = false;

      for(int i = l; i + 2 <= r; i++) {

          if(esPrimo(i) && esPrimo(i+2)){

              twins = true;

              printf("%d %d**\n**",i, i+2);

              break;

          }

      }

      if(!twins) {

          printf("%d**\n**",-1);

        }

    }

 return 0;

}