**Find Prime numbers in a range**

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

Generate all prime numbers between two given numbers.

**Input:**

The first line contains t, the number of test cases (less then or equal to 70). Followed by t lines which contain two numbers m and n (1 <= m <= n <= 1000000000, n-m<=100000) separated by a space.  
**Output:**

For every test case print all prime numbers p such that m <= p <= n, space separated. Separate the answers for each test case by an empty line.  
**Example:**

**Input:**

2

1 10

3 5

**Output:**

2 3 5 7

3 5

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

<http://www.practice.geeksforgeeks.org/problem-page.php?pid=374>

#include <stdio.h>

#include <iostream>

#include <math.h>

#define ll long long int

using namespace std;

int main() {

    //int n = 100;

    int t;

    scanf("%d", &t);

    while(t--) {

        ll n,m;

        scanf("%lld %lld", &m, &n);

        bool prime[n+1];

        //memset(prime, true, sizeof(prime));

        for(int i =0; i<n+1; i++)

            prime[i] = true;

        for (ll p=2; p\*p<=n; p++)

        {

            // If prime[p] is not changed, then it is a prime

            if (prime[p] == true)

            {

                // Update all multiples of p

                for (ll i=p\*2; i<=n; i += p)

                    prime[i] = false;

            }

        }

        // Print all prime numbers

        for (ll p=2; p<=n; p++)

           if (prime[p] && p>= m && p<=n)

              cout << p << " ";

       printf("**\n**");

   }

 return 0;

}