```cpp
/* Prime numbers between a & b using Sieve of Erotosthenes */

#include <stdio.h>
#include <string.h>
#include <map>
#include <iostream>
#include <string>
#include <vector>
#include <iterator>
#include <algorithm>
#include <stdio.h>
#include <stdlib.h>
#include <string>
#define gc getchar_unlocked
#define MOD 1000000009
using namespace std;
inline int getint()
    {
        int num = 0;
        char c = getchar_unlocked();
        int flag = 0;
        while(!((c>='0' & c<='9') || c == '-'))
            c=getchar_unlocked();
        if(c == '-')
        {
            flag = 1;
            c=getchar_unlocked();
        }
        while(c>='0' && c<='9')
        {
            num = (num<<1)+(num<<3)+c-'0';
            c=getchar_unlocked();
        }
        if(flag==0)
            return num;
        else
            return -1*num;
    }

int main()
{
    int isprime[1000000]={0};
    int i,j,n=1000000,a,b,k,t,k1,zz;
    isprime[0]=1;
    isprime[1]=1;
    for(i=2; i<=n; i++)
    {
        if(isprime[i]==0)
        for(j=2*i; j<=n; j+=i)
        {
            isprime[j]=1;
        }
    }

    //array to store the count;
    int ct[1000001]={0};
    for(i=1; i<n; i++)
    {
        ct[i]=ct[i-1];
        if(!isprime[i])
            ct[i]++;
    }

    for(i=1; i<13; i++)
```

```
65          cout<<"i:"<<i<<" "<<ct[i]<<endl;
66
67      t = getint();
68      while(t--)
69      {
70          a = getint();
71          b = getint();
72          cout<<(ct[b]-(ct[a-1]))<<endl;
73
74      }
75  }
76  //1 100000 1 u need to get 9700.
```