Question:

You are given an integer **N**.
You have to print **number of prime numbers** which are less than or equal to N

Input -
First line will contain an integer 'T' (number of test cases).
For each test case there is an integer 'N'.

output -
for each test case print required answer.

constraints :
1<=T<=100000
2<=N<=10000000

Pre-requisites:

Sieve Of Eratosthenes.

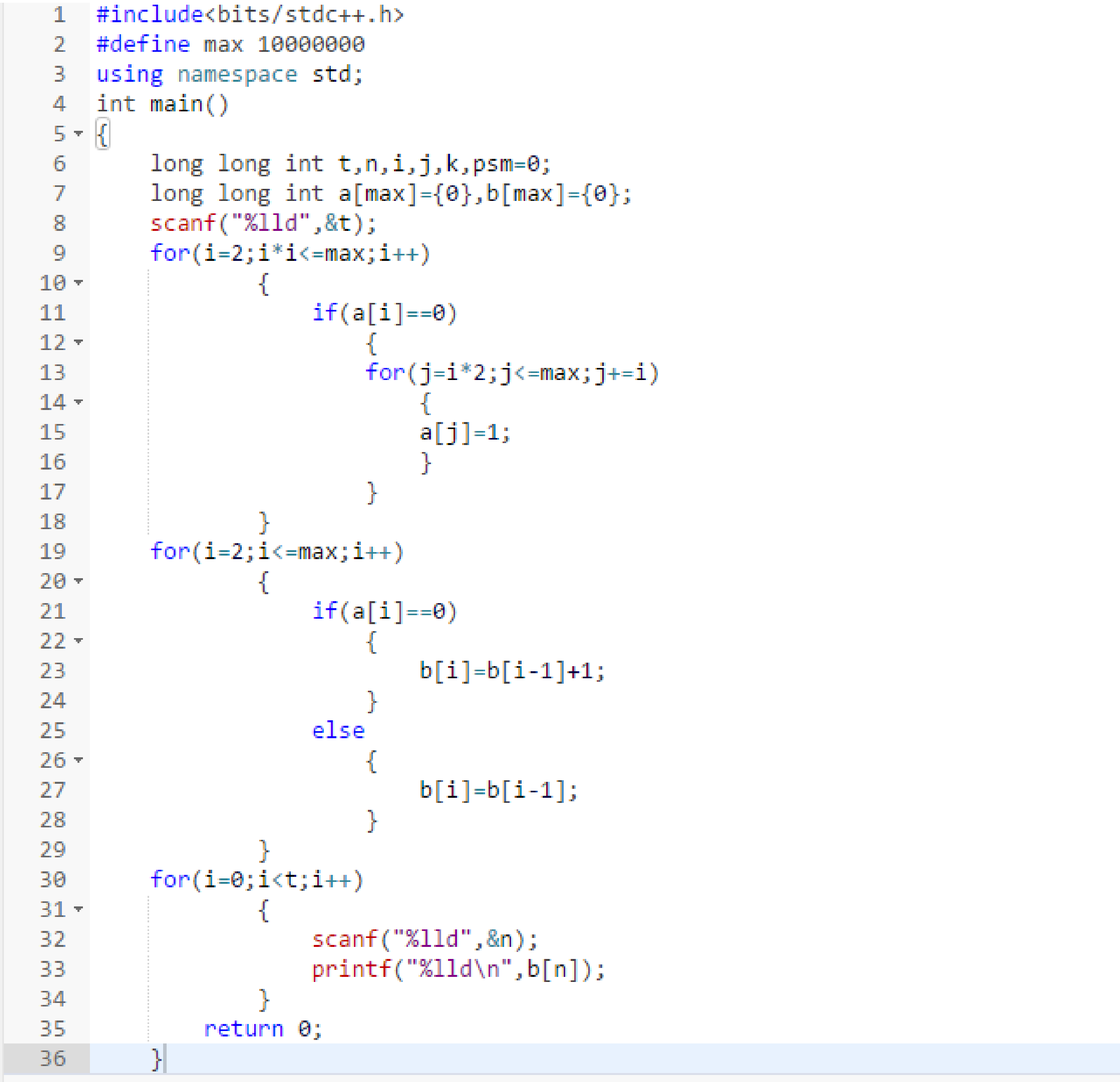
Editorial:

First you need to store a response for each number(Till **10000000** in this question) using sieve. The response is 1 for non-prime and 0 for prime.

Now in other loop, first check the number is prime or not. If number is prime than store the value of last number+1 in it. And if is not prime than store the value of last number in it. With this process we have stored the prime serial number(number of primes till that number) of each number in the corresponding array element.

Now if we have input N, the number of primes less than or equal to N will be the value of array element of index N.

Solution in c++ :



```
#include<bits/stdc++.h>
#define max 10000000
using namespace std;
int main()
{
    long long int t,n,i,j,k,psm=0;
    long long int a[max]={0},b[max]={0};
    scanf("%lld",&t);
    for(i=2;i*i<=max;i++)
            {
                if(a[i]==0)
                    {
                    for(j=i*2;j<=max;j+=i)
                        {
                        a[j]=1;
                        }
                    }
            }
    for(i=2;i<=max;i++)
            {
                if(a[i]==0)
                    {
                        b[i]=b[i-1]+1;
                    }
                else
                    {
                        b[i]=b[i-1];
                    }
            }
    for(i=0;i<t;i++)
            {
                scanf("%lld",&n);
                printf("%lld\n",b[n]);
            }
        return 0;
    }
```