**Number Of Factors**

A number is called n-factorful if it has exactly n distinct prime factors. Given positive integers a, b, and n, your task is to find the number of integers between a and b, inclusive, that are n-factorful. We consider 1 to be 0-factorful.

#include<iostream>

#include<vector>

using namespace std;

int f[1000000];

//sieve for factors

void sieve(int n) {

for(int i=0;i<n+1;i++)

f[i]=0;

for(int i=2;i<=n;i++) {

if(f[i]==0){

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

f[j]++;

}

}

}}

int main() {

int t;

cin>>t;

vector<pair<int,int>> m;

int \*g=new int[t];

for(int i=0;i<t;i++) {

int a,b,n;

cin>>a>>b>>n;

pair<int,int> p;

p.first=a;

p.second=b;

g[i]=n; //storing n in array

m.push\_back(p); //storing a and b in vector

}

int max=m[0].second; //finding max b of array

for(int i=1;i<t;i++) {

if(m[i].second>max)

max=m[i].second;

}

sieve(max);

for(int i=0;i<t;i++) {

pair<int,int> p=m[i];

int d=p.first;

int e=p.second;

int count=0;

for(int j=d;j<=e;j++) {

if(f[j]==g[i])

count++;

}

cout<<count<<endl;

}

return 0;

}