

Number Theory Lecture 2

Saturday, 6 July 2024 8:12 PM

<https://codeforces.com/problemset/problem/17/A>

b/w 2 and 'N' there at least 'k' integers

2 and 10 ^{N=10}

$$a = \underbrace{(P_1 + P_2)}_{\text{consecutive}} + 1$$

$$2 + 3 + 1 = 6$$

$$3 + 5 + 1 = 9$$

$$5 + 7 + 1 = 13$$

$$7 + 11 + 1 = 19$$

$$11 + 13 + 1 = 25$$

$$13 + 17 + 1 = 31$$

$$17 + 19 + 1 = 37$$

$$19 + 23 + 1 = 43$$

$$2 - 45$$

$$\boxed{5}$$

Constraints:-

$$2 \leq n \leq 1000$$

$$0 \leq k \leq 1000$$

$$\bigcirc = \underbrace{\square + \square} + 1$$



n

2-n

Step 2:- Build a vector of a primes till 1000 ✓ $O(n)$

primes

$a[p_1 + p_2 + 1] = \text{True}$

$p_1 + p_2 + 1$

Step 4:- Count the number of elements b/w 2 and n which are True in this array

$\hookrightarrow C[i] \geq k \rightarrow \text{True/Yes}$

other $\rightarrow \underline{\underline{\text{No}}}$

Pr = 2, 3, 5, 7, 11, 13, 17, ... $O(n)$

1		13			19			31			1000	
F	F	-	-	-	T	F	F	T	F	T	1	0

②

①

2	3	5	7	11
i	i+1	i	i+1	

```
import math
n = 1000
# Step 1
pr = [True for i in range(n+1)]
pr[0] = pr[1] = False
for i in range(2, int(math.sqrt(n))+1):
    if pr[i]:
        for j in range(i*i, n+1, i):
            pr[j] = False

#Step 2
prs = []
for i in range(2, n+1):
    if pr[i]:
        prs.append(i)

#Step 3
cb = [False for i in range(n+1)]
for i in range(len(prs)-1):
    ele = prs[i] + prs[i+1] + 1
    if ele > n:
        break
    if pr[ele]:
        cb[ele] = True

#Step 4
n,k = map(int, input().split())
c = 0
for i in range(2, n+1):
    if cb[i]:
        c += 1
if c >= k:
    print('YES')
else:
    print('NO')
```

Segmented Sieve

Q. Find all prime numbers between N and M .

Constraints:- ① $1 \leq N < M \leq 10^{11}$

② $1 \leq M - N \leq 10^6$

<https://www.spoj.com/problems/PRIME1/>

N, M

$M - N$ is
small

$\sim 10^6$

$\sim 10^5$

$2 \rightarrow \sqrt{M}$

Sieve (M)

M is large

$10^{11}, 10^{12}$

$O(M \cdot \log \log M)$

$int \rightarrow 10^9$

$ull \rightarrow 10^{18}$

if no. is prime

\rightarrow you mark all its multiples

$\sqrt{N} \rightarrow \sqrt{M}$

→ you mark all its multiples

$N = 1000$
 $M = 1040$

$\sqrt{N} \rightarrow \sqrt{M}$

Sieve(N) (\sqrt{N})

$(2, 3, 5, \dots, \sqrt{M})$

$N+0 \quad N+1$
 $N \quad N+1 \quad N+2$



Step 1:- Find all the prime nos b/w 2 and \sqrt{M}
 \hookrightarrow Sieve(\sqrt{M})
 $O(\sqrt{M} \log \log \sqrt{M}) \sim O(\sqrt{M})$

Step 2:- Create augmented array of size $M-N+1$, where elements map to values $N \rightarrow M$.

Step 3:- Mark all the elements which are multiples of prime numbers detected in step 1

✓ - If you want to mark k, in the array you mark the index $k-N$

• what is the starting point?

N, M

$P \times P$

(P) ✓

You want the 1st multiple of P b/w N and M floor

$N=1000, M=1040$

$P=7$ $\frac{1000}{7} = \underline{142.85714}$

$\lceil \frac{1000}{7} \rceil \times 7$

$143 \times 7 = \underline{1001}$

$$\left\{ \begin{array}{l} x = N/P * P \\ \text{if } (x < N) \\ \quad x += P; \end{array} \right\}$$

$$\left\{ \begin{array}{l} \text{if } (x < P * P) \\ \quad x = P * P; \end{array} \right\}$$

$N=1, M=100$

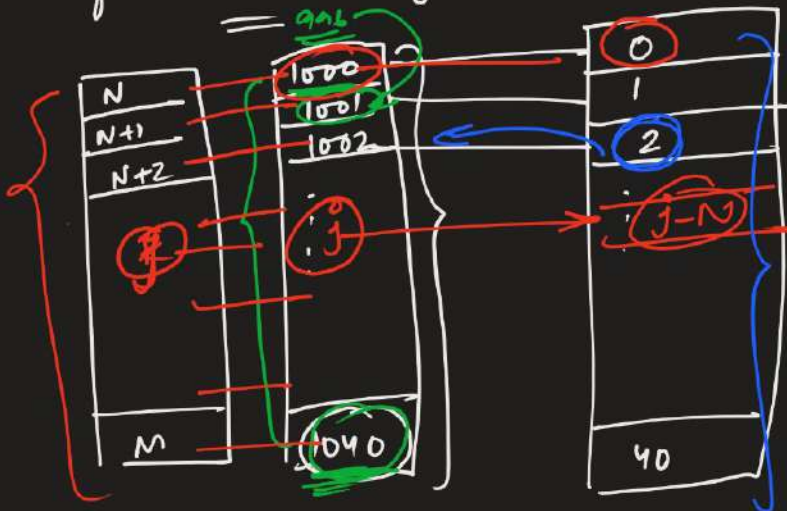
$N=1, M=100$

$3 \rightarrow 3 \times 3$

$P=5$

1	5	25	100
---	---	----	-----

Augmented array



5
6
7
.
.

2, 3, 5, 7, ...

$x = \frac{N}{P} \times i$
 if $(x < N)$
 $x += i$
 $i = 7$
 $7 \times 7 = 49$
 $\frac{996}{7} \times 7$
 integer division

$x = \max(i \times i, x)$

```

#include<bits/stdc++.h>
using namespace std;
#define endl '\n'
#define FOR(i,a,b) for(int i=(a); i<(b); i++)
#define FORk(i,a,b,k) for(int i=(a); i<(b); i+=k)
#define RFOR(i,a,b) for(int i=(a); i>=(b); i--)
#define RFORk(i,a,b,k) for(int i=(a); i>=(b); i-=k)
#define pb push_back
typedef vector<int> vi;
typedef vector<string> vs;
typedef long long int ll;
typedef unsigned long long int ull;
typedef vector<ll> vll;
typedef vector<ull> vull;
const int msm = 31623;
bool pr[msm+1], p[100001];
vi prs;
void sieve() {
    // Step 1
    FOR(i, 2, msm+1)
        pr[i] = true;
    for(int i=2; i*i<=msm; i++)
        if(pr[i])
            FORk(j, i*i, msm+1, i)
                pr[j] = false;
    FOR(i, 2, msm+1)
        if(pr[i])
            prs.pb(i);
}

```



```

void seg_sieve(int N, int M) {
    // Step 2 & 3
    FOR(i,0,M-N+1) p[i] = true;
    if(N==1) p[0] = false; // Border case
    int sm = int(sqrt(M));
    for(auto i: prs) {
        if(i>sm)
            break;
        int x = (N/i)*i;
        if(x < N)
            x += i;
        if(x < i*i)
            x = i*i;
        FORk(j, x, M+1, i) {
            p[j-N] = false;
        }
    }
    FOR(i, 0, M-N+1)
        if(p[i])
            cout << N+i << endl;
}

void solve() {
    int n,m;
    cin >> n >> m;
    seg_sieve(n,m);
    cout << endl;
}

int main() {
    ios_base::sync_with_stdio(false);
    cin.tie(NULL);
    cout.tie(NULL);
    int t = 1;
    cin >> t;
    sieve();
    while(t--) {
        solve();
    }
    return 0;
}

```