

## 1. Задача 2.G

Строка называется бинарной, если она состоит только из символов «0» и «1». Строка  $v$  называется подстрокой строки  $w$ , если она имеет ненулевую длину, и ее можно прочитать, начиная с некоторой позиции, в строке  $w$ . Например, у строки «010» есть шесть подстрок: «0», «1», «0», «01», «10», «010». Две подстроки считаются различными, если их позиции вхождения различны. Другими словами, каждую подстроку нужно учитывать столько раз, сколько она встречается.

Дана бинарная строка  $s$ . Ваша задача — найти количество ее подстрок, содержащих ровно  $k$  единиц.

№	Когда	Кто	Задача	Язык	Вердикт	Время	Память
<a href="#">45919024</a>	2018-11-18 18:27:00	<a href="#">tamarinvs19</a>	<a href="#">G - Еще одна строковая задача</a>	PyPy 3	Полное решение	436 мс	15100 КБ
<a href="#">45918969</a>	2018-11-18 18:24:57	<a href="#">tamarinvs19</a>	<a href="#">G - Еще одна строковая задача</a>	Python 3	Превышено ограничение времени на тесте 39	2000 мс	10600 КБ

```
k = int(input())
ss = input()
m = len(ss)

zeros = []
past = '1'
for s in ss:
    if past == '1' and s == '1':
        zeros.append(0)
    elif past == '1' and s == '0':
        zeros.append(1)
    elif past == '0' and s == '0':
        zeros[-1] += 1
    past = s
if ss[-1] == '1':
    zeros.append(0)
res = 0
if k != 0:
    i = 0
    while i + k < len(zeros):
        res += (zeros[i]+1)*(zeros[i+k]+1)
        i += 1
elif k == 0:
    i = 0
    while i + k < len(zeros):
        res += int(0.5*zeros[i]*(zeros[i+k]+1))
        i += 1
print(res)
```

## 2. Задача 6.H

Известно, что простыми называются целые положительные числа, у которых ровно два различных положительных делителя. По аналогии назовем целое положительное число  $t$  T-простым, если у  $t$  ровно три различных положительных делителя.

Вам дан массив, состоящий из  $n$  целых положительных чисел. Для каждого из них определите, является ли оно T-простым или нет.

№	Когда	Кто	Задача	Язык	Вердикт	Время	Память
<a href="#">45975633</a>	05:23:37	<a href="#">tamarinvs19</a>	<a href="#">H - T-простые числа</a>	Python 3	Превышено ограничение времени на тесте 33	2000 мс	7600 КБ
<a href="#">45975599</a>	05:22:32	<a href="#">tamarinvs19</a>	<a href="#">H - T-простые числа</a>	PyPy 3	Полное решение	1590 мс	19600 КБ

```
def is_prime(n):
    if int(n**0.5)**2 == n:
        n = int(n**0.5)
    else:
        return False
    a = 2
    res = True
    if n == 1:
        res = False
    else:
        while a**2 <= n:
            if n % a == 0:
```

```

        res = False
        break
    a += 1
return res

n = int(input())
ans = {True: 'YES', False: 'NO'}
xs = [ans[is_prime(x)] for x in map(int, input().split())]

print('\n'.join(map(str, xs)))

```

### 3. Задача 6.Е !!

Задан набор из всех целых чисел от  $l$  до  $r$  включительно,  $l < r$ ,  $(r-l+1) \leq 3 \cdot 10^5$  и  $(r-l)$  всегда нечетно.

Вы хотите разделить эти числа на ровно  $\frac{r-l+1}{2}$  пар таким образом, чтобы в каждой паре  $(i, j)$  наибольший общий делитель  $i$  и  $j$  равен 1. Каждое число должно встретиться ровно в одной паре.

Выведите полученные пары или сообщите, что решения не существует. Если существует несколько корректных решений, то выведите любое из них.

№	Когда	Кто	Задача	Язык	Вердикт	Время	Память
<a href="#">45976275</a>	05:48:46	<a href="#">tamarinvs19</a>	<a href="#">Е - Взаимно простые пары</a>	Python 3	Полное решение	280 мс	28000 КБ
<a href="#">45976237</a>	05:47:11	<a href="#">tamarinvs19</a>	<a href="#">Е - Взаимно простые пары</a>	PyPy 3	Полное решение	545 мс	49400 КБ

```

l, r = map(int, input().split())
print('YES')
ans = '{} {}'
pairs = [ans.format(l + 2*i, l + 2*i + 1) for i in range((r - l + 1)//2)]
print('\n'.join(pairs))

```

### 4. Задача 6.D

По введенным числам  $A$  и  $B$  вывести все простые числа в интервале от  $A$  до  $B$  включительно.

<a href="#">45967432</a>	01:21:24	<a href="#">tamarinvs19</a>	<a href="#">D - Решето Эратосфена</a>	Python 3	Полное решение	452 мс	4600 КБ
<a href="#">45967383</a>	01:20:23	<a href="#">tamarinvs19</a>	<a href="#">D - Решето Эратосфена</a>	PyPy 3	Полное решение	186 мс	6300 КБ

```

a, b = map(int, input().split())
a, b = min(a, b), max(a, b)
is_comp = [False] * (b + 1)
for i in range(2, b+1):
    if not is_comp[i-2]:
        j = i ** 2
        while j <= b:
            is_comp[j-2] = True
            j += i
p = [i+2 for i in range(len(is_comp)) if not is_comp[i] and i+2 >= a]
print(' '.join(map(str, p)))

```

### 5. Задача 5.C

Дано  $n$  отрезков на числовой прямой и  $m$  точек на этой же прямой. Для каждой из данных точек определите, скольким отрезкам они принадлежат. Точка  $x$  считается принадлежащей отрезку с концами  $a$  и  $b$ , если выполняется двойное неравенство  $\min(a, b) \leq x \leq \max(a, b)$ .

<a href="#">46070362</a>	2018-11-22 18:59:40	<a href="#">tamarinvs19</a>	<a href="#">D - Дорешивание</a>	Python 3	Полное решение	967 мс	19200 КБ
<a href="#">45477475</a>	2018-11-09 16:08:34	<a href="#">tamarinvs19</a>	<a href="#">C - Точки и отрезки</a>	PyPy 3	Полное решение	1294 мс	22400 КБ

```

from sys import stdin
def association_segments(events):
    i = 1
    last = -1
    c = 0
    events.sort()
    res = dict()
    for x, t in events:
        if t == -1:
            c += 1
        elif t == 0:

```

```

        res[x] = c
    else:
        c -= 1
    return res

n, m = map(int, input().split())
events = []
for _ in range(n):
    l, r = map(int, stdin.readline().strip().split())
    if l > r:
        l, r = r, l
    events.append((l, -1))
    events.append((r, 1))
points = [(x, 0) for x in map(int, input().split())]
events += points
res = association_segments(events)
for p in points:
    print(res[p[0]], end=' ')

```

## 6. Задача 5.D

Как известно, после обеда в ЛКШ проходит много интересных мероприятий, но все равно каждый ЛКШонок старается хотя бы ненадолго заглянуть в комповник, чтобы дорешать задачи, не сделанные во время практики.

В этом году погода стоит особо жаркая, поэтому в комповнике очень душно и важно следить за тем, чтобы в комповнике не находилось одновременно очень много школьников. Поэтому завуч записал время прихода и ухода из комповника каждого ЛКШонка.

Теперь завуч хочет узнать, сколько ЛКШат встретил в комповнике каждый ЛКШонок.

<a href="#">46070362</a>	2018-11-22 18:59:40	<a href="#">tamarinvs19</a>	<a href="#">D - Дорешивание</a>	Python 3	<a href="#">Полное решение</a>	967 мс	19200 КБ
<a href="#">45525712</a>	2018-11-10 18:34:21	tamarinvs19	<a href="#">D - Дорешивание</a>	PyPy 3	<a href="#">Полное решение</a>	1606 мс	13400 КБ

```

n = int(input())
xs = []
for i in range(1, n+1):
    y = list(map(int, input().split()))
    xs.append((y[0], -i))
    xs.append((y[1], i))

xs.sort()
count_open = 0
count_close = 0
ys = [0]*(n+1)
for x, t in xs:
    if t < 0:
        ys[abs(t)] -= count_close
        count_open += 1
    else:
        ys[abs(t)] += count_open
        count_close += 1
for y in ys[1:]:
    print(y-1)

```

## 7.