

Вариант 1

1 Петя написал на доске все двузначные натуральные числа и напротив каждого из чисел – квадрат произведения его цифр. Вася быстро сложил все числа, написанные на доске. Какой результат получил Вася?

2 Найти наибольшее нечетное число, не превосходящее 500, имеющее нечетное число делителей, включая 1 и самого числа.

3. На стороне  $BC$  остроугольного треугольника  $ABC$  как на диаметре построена окружность радиуса 4, пересекающая стороны  $AB$  и  $AC$  в точках  $M$  и  $N$  соответственно.

Найти длину отрезка  $MN$ , если косинус угла при вершине  $A$  треугольника  $ABC$  равен  $\frac{1}{8}$ .

Ответы и решения

**Задача 1** Сумма  $S_1$  произведений квадратов цифр всех двузначных чисел равна:

$$S_1 = (1^2 + 2^2 + 3^2 + \dots + 9^2) \cdot (1^2 + 2^2 + 3^2 + \dots + 9^2).$$

Воспользуемся формулой:  $1^2 + 2^2 + 3^2 + \dots + n^2 = \frac{(2n+1)n(n+1)}{6}$ . Получим:

$$1^2 + 2^2 + 3^2 + \dots + 9^2 = \frac{19 \cdot 9 \cdot 10}{6} = 285.$$

Тогда  $S_1 = 285^2 = 81225$ .

Сумма  $S_2$  всех двузначных чисел равна:  $S_2 = 10 + 11 + 12 + \dots + 98 + 99 = 109 \cdot 45 = 4905$ .

Сложив  $S_1$  и  $S_2$ , получим результат Васи:  $S_1 + S_2 = 81225 + 4905 = 86130$ .

**Ответ:** 86130.

**Задача 2.** Пусть искомое число  $a$  имеет вид:  $a = p_1^{r_1} \cdot p_2^{r_2} \cdot \dots \cdot p_m^{r_m}$ . Здесь  $p_1, p_2, \dots, p_m$  – простые делители,  $r_1, r_2, \dots, r_m$  – их кратности. Так как по условию число  $a$  нечетное, то среди простых делителей нет 2. Число делителей числа  $a$  равно

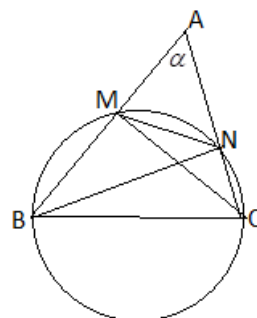
$$Q = (r_1 + 1)(r_2 + 1)(r_3 + 1) \dots (r_m + 1).$$

По условию число делителей нечетное. Из этого следует, все его скобки являются нечетными числами, поэтому числа  $r_1, r_2, \dots, r_m$  – четные. Тогда  $a$  является квадратом нечетного числа, меньшим 500. Наибольшее из таких чисел  $a = 21^2 = 441$ .

**Ответ:** 441.

**Задача 3.** Заметим, что  $BC = 2R = 8$ . Рассмотрим треугольники  $ANM$  и  $ABC$ . Покажем, что они подобные. Во первых, они имеют общий угол  $\angle BAC$  (обозначим его  $\alpha$ ). Во вторых, так как четырехугольник  $BMNC$  вписанный, то  $\angle ABC = \angle MNA$ . Запишем коэффициент подобия:

$$k = \frac{AN}{AB} = \frac{AB \cos \alpha}{AB} = \cos \alpha = \frac{1}{8}.$$



Тогда  $MN = BC \cdot \cos \alpha = 8 \cdot \frac{1}{8} = 1$ .

**Ответ:** 1.

### Вариант 2

1. Петя написал на доске все двузначные натуральные числа, в десятичной записи которых используются только четные цифры и ноль, а напротив каждого из этих чисел – квадрат произведения его цифр. Вася быстро сложил все числа, написанные на доске. Какой результат получил Вася?

**Ответ:** 15480.

2. Найти наименьшее нечетное число, превосходящее 750, имеющее нечетное число делителей, включая 1 и самого числа.

**Ответ:** 841.

3. На стороне  $BC$  остроугольного треугольника  $ABC$  как на диаметре построена окружность радиуса 6, пересекающая стороны  $AB$  и  $AC$  в точках  $M$  и  $N$  соответственно.

Найти длину отрезка  $MN$ , если косинус угла при вершине  $A$  треугольника  $ABC$  равен  $\frac{1}{3}$ .

**Ответ:** 4.

### Вариант 3

1. Петя написал на доске все двузначные натуральные числа, в десятичной записи которых используются только нечетные цифры и ноль, а напротив каждого из этих чисел – квадрат произведения его цифр. Вася быстро сложил все числа, написанные на доске. Какой результат получил Вася?

**Ответ:** 28850.

2. Найти наибольшее нечетное число, не превосходящее 1000, имеющее нечетное число делителей, включая 1 и самого числа.

**Ответ:** 961.

3. На стороне  $BC$  остроугольного треугольника  $ABC$  как на диаметре построена окружность радиуса 5, пересекающая стороны  $AB$  и  $AC$  в точках  $M$  и  $N$  соответственно. Найти длину отрезка  $MN$ , если косинус угла при вершине  $A$  треугольника  $ABC$  равен  $\frac{1}{5}$ .

**Ответ:** 2.

### Вариант 4

1. Петя написал на доске все трехзначные натуральные числа и напротив каждого из чисел – квадрат произведения его цифр. Вася быстро сложил все числа, написанные на доске. Какой результат получил Вася?

**Ответ:** 23643575.

2. Найти наименьшее нечетное число, превосходящее 1250, имеющее нечетное число делителей, включая 1 и самого числа.

**Ответ:** 1369.

3. На стороне  $BC$  остроугольного треугольника  $ABC$  как на диаметре построена окружность радиуса 8, пересекающая стороны  $AB$  и  $AC$  в точках  $M$  и  $N$  соответственно. Найти длину

отрезка  $MN$ , если косинус угла при вершине  $A$  треугольника  $ABC$  равен  $\frac{1}{2}$ .

**Ответ:** 8.

## Критерии проверки работ

Всероссийский конкурс научных работ школьников «Юниор», профиль «Инженерные науки», Олимпиада по математике, Заключительный этап, 9 класс

Во всех задачах ответ без решения – 0 б.

### Задача 1:

- 1 б – Записал сумму квадратов произведений заданных чисел, но не смог ее вычислить или вычислил с ошибкой.
- 2б – Вычислил сумму квадратов произведений заданных чисел, но забыл добавить сумму самих чисел.
- 3 б – Задача решена верно.

### Задача 2

- 1 б- Записал формулы представления числа в виде произведений степеней простых чисел и числа делителей через эти степени.
- 2 б – Задача решена с мелкими недочётами или арифметической ошибкой, не влияющей на ход решения.
- 3 б - Задача решена верно.

### Задача 3

- 1 б –Правильно построил чертёж, нашел равные углы и подобные треугольники.
- 2 б – Задача решена с мелкими недочётами или арифметической ошибкой, не влияющей на ход решения.
- 3 б - Задача решена верно.

## 1. А. Делим на 9

### 1.1 Условие

В МИФИ происходят перестановки в расписании, отчего у вас попросили узнать следующее:

Допустим, у вас есть массив  $a$  длины  $n$ . Он состоит из целых положительных чисел. Вам необходимо определить, возможно ли переставить его элементы таким образом, чтобы произведение любых двух соседних элементов делилось нацело на 9.

### 1.2 Входные данные

Первая строка ввода содержит одно целое число  $n$  ( $1 \leq n \leq 2 \cdot 10^5$ ). Вторая строка содержит  $n$  целых положительных чисел  $a_1, a_2, \dots, a_n$  ( $1 \leq a_i \leq 10^9$ ).

### 1.3 Вывод

Выведите Yes, если это возможно, и No в противном случае

### 1.4 Пример входных данных

**Sample Input:**

7

8 8 8 9 8 7 8

**Sample Output:**

No

**Sample Input:**

7

7 7 9 9 7 9 7

**Sample Output:**

Yes

### 1.5 Решение

Посчитаем, сколько в массиве чисел кратных 9 (числа первого типа), сколько кратных 3, но не кратных 9 (числа второго типа) и остальных чисел (третий тип).

Рассмотрим случай, когда нет чисел второго типа. Тогда достаточно поставить поочередно числа первого и третьего типа. Если нам хватает чисел первого типа для такой расстановки – ответ “Yes”, иначе это невозможно.

Рассмотрим более сложный случай – у нас есть числа второго типа. Очевидно, что нам нет смысла специально ставить рядом числа первого типа или ставить рядом первый и второй тип. Но тогда числа кратные 3 (второго типа) должны идти друг за другом, иначе не будет выполняться условие задачи. Получается, что достаточно их поставить, например, в начало массива. После этого задача сводится к более простой, уже рассмотренной нами. Непосредственно в коде строить такой массив не нужно, достаточно проверить несколько условий. Решение за  $O(n)$ .

```
type1 = 0
type2 = 0
type3 = 0
n = int(input())
a = list(map(int, input().split()))
for i in range(n):
    if a[i] % 9 == 0:
        type1 += 1
    elif a[i] % 3 == 0:
        type2 += 1
    else:
        type3 += 1
if type2 == 0:
    if n // 2 <= type1:
        print("Yes")
    else:
        print("No")
else:
    if (n - type2 + 1) // 2 <= type1:
        print("Yes")
    else:
        print("No")
```

## 2. В. Играем в игры

### 2.1 Условие

У мифиста есть  $n$  игр. В  $i$ -ой игре есть  $a[i]$  число сессий, которые он должен в нее отыграть. Пусть дано целое число  $b$ , определяющее, сколько удовольствия он получит. Тогда за время прохождения всех сессий  $i$ -ой игры его удовольствие изменится следующим образом:

- 1) За время первой сессии он получит  $b$  единиц удовлетворения;
- 2) За вторую  $b-1$ ;
- 3) За третью  $b-2$  и так далее, пока все сессии не будут отыграны.

Соответственно, после отыгрывания очередной сессии удовольствие мифиста может начать уменьшаться. Также он обязан отыграть все  $a[i]$  сессий.

Необходимо узнать, хватит ли сил мифисту, чтобы сдать сессию. Для этого обработайте  $q$  запросов вида: даны числа  $L$  и  $b$ . Найдите такое число  $R$  ( $L \leq R$ ), что он отыграет все сессии в каждой игре от  $L$  до  $R$  включительно и получит наибольшее удовольствие. Обратите внимание, что ответ может быть очень большим, поэтому используйте подходящий целочисленный тип данных в вашем языке (`long long` в C++, `Long` в Java и т.д.).

### 2.2 Входные данные

Первая строка ввода содержит одно целое число  $n$  ( $1 \leq n \leq 5000$ ).

Вторая строка содержит  $n$  целых чисел  $a_1, a_2, \dots, a_n$  ( $1 \leq a_i \leq 10^6$ ).

Третья строка содержит одно целое число  $q$  ( $1 \leq q \leq 5000$ ).

Следующие  $q$  строк содержат по два целых числа  $L$  и  $b$  ( $1 \leq L \leq n, 1 \leq b \leq 10^6$ ) - описания каждого запроса.

### 2.3 Вывод

Выведите  $q$  целых чисел, по одному на каждой строке.  $i$ -я строка должна содержать лучшее значение  $R$  для  $i$ -ого запроса. Если существует несколько возможных ответов, выведите наименьший из них.

### 2.4 Пример входных данных

**Sample Input:**

```
8
13811633
4
11
12
13
41
```

**Sample Output:**

```
1
2
8
5
```

## 2.5 Решение

Рассмотрим 1 игровую сессию, где число  $b$  определяет уровень удовольствия и где в игру надо отыграть  $a$  сессий. Тогда получится, что мы получим по очереди  $b, b-1, \dots, b - a + 1$  удовольствия. Это сумма арифметической прогрессии с разностью 1, первым элементом  $b - a + 1$  и количеством элементов равным  $a$ . Тогда сумма такой прогрессии равна  $(b - a + 1 + b) * a / 2 = (2 * b * a - a * a + a) / 2$ . Рассмотрим обработку запроса. Нам достаточно идти указателем по массиву и поддерживать текущее удовольствие. Если при игре во все игры до текущей позиции включительно удовольствие стало строго больше предыдущего максимума, то это оптимальный ответ на данный момент. Так можно сделать, просто пройдясь по массиву для каждого запроса. Решение за  $O(n*q)$ .

```
n = int(input())
a = list(map(int, input().split()))
q = int(input())
while q:
    l, b = map(int, input().split())
    l -= 1
    su = 0
    ans = -1e18
    R = -1
    for i in range(l, n):
        su += (a[i] + 2 * a[i] * b - a[i] * a[i]) // 2
        if su > ans:
            R = i + 1
            ans = su
    print(R)
    q -= 1
```

### 3. С. Обьедаемся пирожками

#### 3.1 Условие

Сегодня у Владислава знаменательный день: он пишет финальный этап олимпиады "ЮНИОР". В силу этого ему нужно хорошо подкрепиться пирожками.

Так как в МИФИ за последнее время открылось очень много кафетериев, теперь можно приобрести целых  $n$  видов пирожков. Влад хочет ими пообедать, поэтому Вы хотите ему в этом помочь.

Влад очень не любит  $k$  чисел. Он хочет купить какой-то непустой набор пирожков так, чтобы число пирожков в нем не было его нелюбимым. Но также Влад любит разнообразие, поэтому он не хочет покупать более 1 пирожка каждого вида.

Помогите ему найти число различных наборов пирожков, удовлетворяющих описанным выше условиям.

Так как число способов может быть крайне большим, выведите ответ по модулю 1000000007.

Обратите внимание, что у Влада нелюбимых чисел не больше 20.

#### 3.2 Входные данные

Первая строка ввода содержит два целых числа  $n$  ( $1 \leq n \leq 10^9$ ) и  $k$  ( $1 \leq k \leq \min(n, 20)$ ).

Вторая строка содержит  $k$  целых нелюбимых различных чисел  $a_1, a_2, \dots, a_k$  ( $1 \leq a_i \leq \min(n, 2 \cdot 10^5)$ ),  $a_i \neq a_j$  при  $i \neq j$ .

#### 3.3 Вывод

Выведите число способов, которыми он может набрать себе пирожки на обед по модулю 1000000007.

#### 3.4 Пример входных данных

**Sample Input:**

3 1

2

**Sample Output:**

4

#### 3.5 Решение

Заметим, что, если переформулировать условие, то нужно выбрать из  $n$  чисел от 1 до  $n$  любой набор из них, размер которого не является нелюбимым числом. Количество всех различных наборов размера  $x$  – это число сочетаний из  $n$  по  $x$ . Есть известная формула, что сумма всех сочетаний из  $n$  по  $0, 1, 2, \dots, n$  равна  $2^n$ . Это число можно быстро посчитать по данному модулю, применив бинарное возведение в степень. Нам нужны все сочетания из  $n$  по  $1, \dots, n$  за исключением  $k$  из них. Получается, что ответом будет являться разность  $2^n$  и сочетаний из  $n$  по  $0$ , из  $n$  по  $a_1$ , из  $n$  по  $a_2$  и т.д. Эти  $\sim 20$  сочетаний можно посчитать по обычной формуле сочетаний, так как  $a_i \leq 2 \cdot 10^5$ . Также нужно помнить, что все операции должны проводиться по данному модулю. Решение за  $O(\log(n) + k \cdot \max(a_i))$ .

```
m = 1000000007
```

```
def pw(a, b):  
    ans = 1  
    while b:  
        if b % 2:  
            ans = (ans * a) % m  
        a = (a * a) % m  
        b //= 2  
    return ans
```

```
n, k = map(int, input().split())  
a = list(map(int, input().split()))  
ans = pw(2, n)  
ans = m - 1 + ans  
ans %= m  
for i in range(k):  
    s = 1  
    s1 = 1  
    for j in range(n - a[i] + 1, n + 1):  
        s *= j  
        s %= m  
    for j in range(2, a[i] + 1):  
        s1 *= j  
        s1 %= m  
    s1 = pw(s1, m - 2)  
    s *= s1  
    s %= m  
    ans += m - s  
    ans %= m  
print(ans)
```