

## Задача А. Задача для второклассника

Имя входного файла: multiply.in  
Имя выходного файла: multiply.out  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

Вам даны два числа. Необходимо найти их произведение.

### Формат входных данных

Входные данные состоят из двух строк, на каждой из которых находится целое одно **целое** число, длина которого не превосходит двухсот пятидесяти тысяч символов.

### Формат выходных данных

Выведите произведение данных чисел.

### Примеры

multiply.in	multiply.out
2	4
2	

## Задача В. Банковское дело

Имя входного файла: `bankcraft.in`  
Имя выходного файла: `bankcraft.out`  
Ограничение по времени: 10 секунд  
Ограничение по памяти: 512 мегабайт

Ося и его банда хотят экспроприировать деньги одного нечестного миллионера.

У них есть следующая проблема. Миллионер хранит свои деньги в банке. Банк использует криптографическую схему с открытым ключом для авторизации своих клиентов. У каждого клиента есть свой собственный публичный ключ, который является многочленом  $P(x)$  над полем остатков по модулю простого числа  $p$ , и приватный ключ — многочлен  $Q(x)$  над тем же самым полем. Приватный ключ считается правильным, если существует многочлен  $R(x)$ , такой, что  $P(x) \cdot Q(x) = 1 + x^m \cdot R(x)$  для некоторого зафиксированного числа  $m$ .

Ося знает многочлен  $P(x)$ , число  $p$  (оно всегда равно 7340033) и число  $m$ , но он не знает приватный ключ. Он предлагает вам оценку «5+» на зачёте, за помощь в нахождении этого ключа. Вы же не можете отказаться от такого щедрого предложения?

### Формат входных данных

В первой строке входного файла находятся два целых числа:  $m$  и  $n$  ( $1 \leq m, n \leq 10^5$ ).  $n$  — степень многочлена  $P(x)$ . Вторая строка содержит  $n + 1$  целое число  $a_i$  ( $0 \leq a_i \leq p - 1$ ) — коэффициенты многочлена  $P(x)$ .  $i$ -ое из них ( $0 \leq i \leq n$ ) — это коэффициент при  $x^i$ .

### Формат выходных данных

Если невозможно найти подходящий многочлен степени менее  $m$ , выведите сообщение «The ears of a dead donkey»<sup>1</sup> (без кавычек). Если решение существует, то выведите  $m$  целых чисел  $b_i$  ( $0 \leq b_i \leq p - 1$ ), являющихся коэффициентами  $Q(x)$ . Если существует несколько вариантов ответа, выведите тот, который вам больше нравится.

### Примеры

<code>bankcraft.in</code>	<code>bankcraft.out</code>
2 1 1 2	1 7340031
4 2 1 0 1	1 0 7340032 0

---

<sup>1</sup>От мёртвого осла уши (англ.)

## Задача С. ДПФ для чайников

Имя входного файла: `dft.in`  
Имя выходного файла: `dft.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 512 мегабайт

Нормированное дискретное преобразование Фурье (ДПФ) для последовательности комплексных чисел  $x_0, x_1, \dots, x_{n-1}$  — это последовательность чисел  $y_0, y_1, \dots, y_{n-1}$ , которая определяется по следующей формуле:

$$y_k = \frac{1}{\sqrt{n}} \sum_{j=0}^{n-1} x_j \cdot \exp\left(\frac{2\pi i j k}{n}\right).$$

Здесь  $i$  обозначает мнимую единицу.

Евклидовой нормой последовательности  $x_0, x_1, \dots, x_{n-1}$  называется выражение, равное

$$\|x\| = \sqrt{|x_0|^2 + |x_1|^2 + \dots + |x_{n-1}|^2}.$$

Вам дан массив из  $n$  чисел, вам требуется ответить на несколько запросов. Каждый запрос описывается отрезком массива  $[l, r]$  и состоит из трех этапов:

1. Заменить отрезок  $[l, r]$  на его дискретное преобразование Фурье.
2. Вывести квадрат евклидовой нормы отрезка  $[l, r]$ .
3. Заменить отрезок  $[l, r]$  на его дискретное преобразование Фурье еще раз.

По определению,  $\exp(i \cdot \varphi) = \cos \varphi + i \cdot \sin \varphi$ , где  $\varphi$  — произвольное вещественное число. Модуль комплексного числа  $|z|$  определяется для числа  $z = a + i \cdot b$  как  $\sqrt{a^2 + b^2}$ .

### Формат входных данных

В первой строке входных данных содержится число  $n$  — длина массива ( $1 \leq n \leq 10^5$ ). В следующей строке содержатся  $n$  целых чисел  $a_i$ , исходные элементы массива ( $|a_i| \leq 10^6$ ). Обратите внимание, что после преобразования числа становятся комплексными. Следующая строка содержит число  $q$  — количество запросов ( $1 \leq q \leq 10^5$ ). Каждая из следующих  $q$  строк описывает один запрос. Каждое описание состоит из двух чисел  $l$  и  $r$  ( $1 \leq l \leq r \leq n$ ).

### Формат выходных данных

Для каждого запроса выведите одно число — искомую сумму квадратов, округленную **вверх** до ближайшего целого числа.

### Примеры

<code>dft.in</code>	<code>dft.out</code>
2	1
1 0	0
3	1
1 1	
2 2	
1 2	

## Задача D. Подмножества наносят ответный удар

Имя входного файла: `combo.in`  
Имя выходного файла: `combo.out`  
Ограничение по времени: 7 секунд  
Ограничение по памяти: 512 мегабайт

Имея множество из  $n$  различных элементов, легко найти количество его различных  $k$ -элементных подмножеств. Это хорошо известный биномиальный коэффициент  $\binom{n}{k} = \frac{n!}{k!(n-k)!}$ .

Однако, данная задача несколько сложнее. Вам дано мультимножество, состоящее из  $n$  (возможно, совпадающих) элементов, и вам требуется ответить на несколько вопросов вида “если бы мы добавили элемент  $x$  в мультимножество, то сколько бы в нем существовало различных  $k$ -элементных подмультимножеств?”. Обратите внимание, что после каждого запроса множество **не изменяется**.

Напомним, что  $k$ -элементное подмультимножество — это любые  $k$  элементов исходного мультимножества (разрешается брать одинаковые элементы). Два мультимножества считаются различными, если какой-либо элемент встречается в них разное число раз.

Так как ответы на запросы могут быть слишком большими, от вас требуется вывести их по модулю 1 051 721 729.

### Формат входных данных

В первой строке входных данных содержится число  $n$  — размер мультимножества ( $1 \leq n \leq 120\,000$ ). В следующей строке содержатся  $n$  чисел  $a_i$ : элементы мультимножества ( $1 \leq a_i \leq n$ ). Обратите внимание, что элементы могут совпадать. В следующей строке содержится число  $q$  — количество запросов ( $1 \leq q \leq 120\,000$ ). Каждый из следующих  $q$  строк описывает запрос, описание запроса состоит из двух чисел  $x$  и  $k$  ( $1 \leq x \leq n$ ,  $1 \leq k \leq n + 1$ ).

### Формат выходных данных

Для каждого запроса выведите одно число — искомое количество подмультимножеств, взятое по модулю 1 051 721 729.

### Примеры

<code>combo.in</code>	<code>combo.out</code>
6	6
1 2 2 3 3 3	7
4	6
1 2	8
2 3	
3 4	
4 5	

### Замечание

При выполнении первого запроса (если мы добавим элемент 1), исходное мультимножество будет выглядеть как 1 2 2 3 3 3 1, таким образом, оно будет содержать 6 различных подмультимножеств размера 2: (1 1), (1 2), (1 3), (2 2), (2 3) и (3 3).

## Задача E. Ним

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 512 мегабайт

Алиса и Боб собираются поиграть в Ним. Сначала они выставляют  $k$  стопок камней, содержащие  $a_1, a_2, \dots, a_k$  камней соответственно. Затем они по очереди ходят, при чём Алиса начинает первой.

В свой ход, игрок выбирает любую непустую кучку и забирает оттуда любое ненулевое количество камней, если на момент хода все кучки пусты, то у игрока нет хода и он проигрывает.

Из особой любви к простым числам, они решили сделать каждое  $a_i$  простым числом от 0 до  $l$ . По данным  $k$  и  $l$  найдите число начальных состояний, дающих победу Бобу, если оба игрока играют оптимально. Выведите ответ по модулю  $10^9 + 7$ .

### Формат входных данных

Единственная строка входных данных содержит два числа:  $k$  и  $l$  ( $1 \leq k \leq 10^9$ ,  $2 \leq l \leq 5 \cdot 10^4$ )

### Формат выходных данных

Выведите по модулю  $10^9 + 7$  число начальных состояний, обеспечивающих выигрыш Бобу.

### Примеры

стандартный ввод	стандартный вывод
3 7	6
4 13	120