



# Общая информация по задачам олимпиады

## Ограничение по памяти

- В задаче A: 256 мегабайт
- В задаче B: 256мегабайт
- В задаче C: 256 мегабайт
- В задаче D: 256 мегабайт
- В задаче E: 512 мегабайт

## Ограничение на размер исходного кода программы

Во всех задачах размер файла с исходным кодом решения не должен превышать 256 КБ.

## Ограничение на посылку решений

По каждой задаче на проверку принимается не более 50 решений.

По каждой задаче участник не может отправить решение более одного раза в течение 30 секунд. Это ограничение не распространяется на последние 15 минут соревнований.

## Система оценки

Каждая задача олимпиады поделена на несколько подзадач. Чтобы набрать баллы по подзадаче, программа должна пройти все тесты этой подзадачи.

За каждую задачу выставляется суммарный балл по всем ее подзадачам. В каждой подзадаче оценивается лучшее решение, то есть за подзадачу выставляется максимальный набранный по ней балл среди всех решений.

## Получение информации о результатах проверки

Чтобы получить информацию о проверке вашего решения, используйте ссылку «Информация о проверке» во вкладке «Решения» в PCMS2 Web Client. По каждой задаче вам будет доступна информация по количеству набранных баллов в каждой подзадаче или результат проверки на первом непройденном тесте.

## Таблица результатов

Во время соревнования доступна текущая таблица результатов. Для доступа к ней используйте ссылку «Результаты» в PCMS2 Web Client. Таблица результатов в PCMS2 Web Client не является окончательной.



## Задача А. Шустрик, Персик и вечная дружба

Ограничение по времени: 1 секунда

В древнем храме спрятано легендарное сокровище, символизирующее вечную дружбу хомяка Шустрика и кота Персика. Шустрик и Персик хотят показать вам это волшебное место. Чтобы попасть туда, нужно открыть врата древнего храма, для этого необходимо выстроить кольцевую мозаику из сакральных символов (цифры 0-9) в правильном порядке.

Мозаика будет считаться выстроенной в правильном порядке, если потенциал строки сакральных символов и количество применений волшебной операции будут как можно меньше. Потенциалом строки называется количество блоков одинаковых цифр, из которых она состоит. Например, у строки 112223 потенциал 3, а у 12233321 — 5. Волшебная операция, которую могут произвести Шустрик и Персик, выглядит так: они могут циклически сдвигать строку вправо или влево на 1. Так как место действительно волшебное, то друзья могут выполнять эту операцию сколько угодно раз.

Помогите Шустрику и Персику посчитать, сколько раз им придётся применить волшебную операцию, чтобы мозаика стала выстроена в правильном порядке. Помните, что мозаика считается выстроенной в правильном порядке только тогда, когда потенциал строки сакральных символов и количество применений волшебной операции минимальны.

### Формат входных данных

В первой строке дано одно целое число  $n$  ( $1 \leq n \leq 10^6$ ).

Во второй строке дана строка  $s$  — мозаика из сакральных символов.

### Формат выходных данных

Выведите количество волшебных операций, которые придётся применить Шустрику и Персику. Если во время волшебных операций друзья делают сдвиги влево, то нужно вывести количество волшебных операций со знаком «-» (без кавычек), если вправо — без знака «-».

### Система оценки

| Группа | Баллы | Дополнительные ограничения     | Необходимые группы |
|--------|-------|--------------------------------|--------------------|
| 1      | 20    | $n \leq 8$                     | —                  |
| 2      | 10    | $n \leq 18$                    | 1                  |
| 3      | 30    | $n \leq 1\,000$                | 1, 2               |
| 4      | 40    | Нет дополнительных ограничений | 1, 2, 3            |

### Примеры

| стандартный ввод | стандартный вывод |
|------------------|-------------------|
| 6<br>112223      | 0                 |
| 9<br>112223111   | -2                |
| 9<br>122333211   | -1                |

### Замечание





## Задача В. Магический круг

Ограничение по времени: 2 секунды

$n$  магов встали в круг и решили провести  $q$  ритуалов.

Пронумеруем магов от 1 до  $n$  по часовой стрелке.  $i$ -й ритуал задействует мага номер  $k$ , если существует такое целое неотрицательное  $x$ , что  $(first_i + step_i \cdot x - 1) \bmod n + 1 = k$ , где  $\bmod$  — это операция взятия остатка от деления. Для лучшего понимания стоит посмотреть на иллюстрацию к примеру.

Изначально у каждого мага было  $10^{18}$  единиц маны.  $i$ -й ритуал изменяет ману задействованных магов на  $cost_i$ . То есть если  $cost_i > 0$ , то мана увеличивается на  $|cost_i|$  единиц, если  $cost_i < 0$ , то — уменьшается на  $|cost_i|$  единиц, а если  $cost_i = 0$ , то — не изменяется.

Определите номер мага с наибольшим количеством маны после всех  $q$  ритуалов. Если таких несколько, то — с минимальным номером.

### Формат входных данных

В первой строке даны два целых числа  $n$  и  $q$  ( $1 \leq n, q \leq 2 \cdot 10^5$ ).

В  $i$ -й из следующих  $q$  строк даны 3 целых числа  $first_i$ ,  $step_i$  и  $cost_i$  ( $1 \leq first_i \leq n$ ;  $1 \leq step_i \leq 10^9$ ;  $-10^9 \leq cost_i \leq 10^9$ ).

### Формат выходных данных

Выведите одно число — номер мага с наибольшим количеством маны после всех  $q$  ритуалов. Если таких несколько, то — с минимальным номером.

### Система оценки

Тесты к этой задаче состоят из десяти групп. Баллы за каждую группу ставятся только при прохождении всех тестов группы и всех тестов необходимых для неё групп.

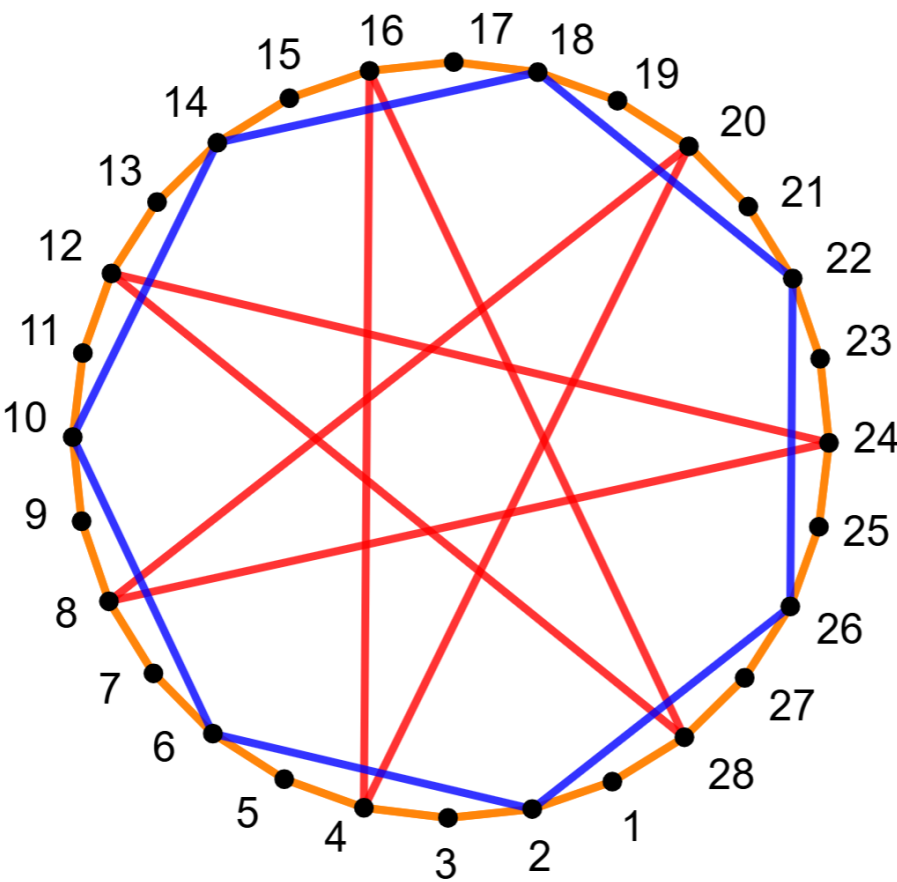
| Группа | Баллы | Доп. ограничения                                                       | Необходимые группы |
|--------|-------|------------------------------------------------------------------------|--------------------|
| 1      | 10    | $n, q \leq 5\,000$ ; $n$ кратно $step_i$ для всех $i$                  | —                  |
| 2      | 10    | $n, q \leq 5\,000$ ; $step_i \leq n$ для всех $i$                      | 1                  |
| 3      | 10    | $n, q \leq 5\,000$                                                     | 1, 2               |
| 4      | 10    | $first_i = first_j$ для всех $i, j$ ; $n$ кратно $step_i$ для всех $i$ | —                  |
| 5      | 10    | $step_i = step_j$ для всех $i, j$ ; $n$ кратно $step_i$ для всех $i$   | —                  |
| 6      | 10    | $n$ кратно $step_i$ для всех $i$                                       | 1, 4, 5            |
| 7      | 10    | $first_i = first_j$ для всех $i, j$ ; $step_i \leq n$ для всех $i$     | 4                  |
| 8      | 10    | $step_i = step_j$ для всех $i, j$ ; $step_i \leq n$ для всех $i$       | 5                  |
| 9      | 10    | $step_i \leq n$ для всех $i$                                           | 1, 2, 4–8          |
| 10     | 10    | —                                                                      | 1–9                |



Пример

| стандартный ввод                   | стандартный вывод |
|------------------------------------|-------------------|
| 28 3<br>28 12 8<br>18 4 4<br>3 1 2 | 4                 |

Иллюстрация





## Задача С. Приятные пути

Ограничение по времени: 1 секунда

Карта леса представляет собой корневое дерево, вершины которого — поляны, рёбра — тропинки между ними, а вес ребра — протоптанность этих тропинок. Все поляны пронумерованы от 1 до  $n$ .

Ёжик Шуршик живёт на поляне, которая находится в вершине-корне под номером 1. Во время каждой своей прогулки ёжик отправляется из вершины 1 и идёт в вершины-листья за самыми вкусными яблоками, но только *приятным путём*. Приятный путь в дереве — путь из вершины-корня в вершину-лист такой, что при каждом спуске ёжик выбирает самую непопулярную дорогу (т.е. спускается по ребру с наименьшим значением протоптанности, а если таких несколько — в вершину с наименьшим номером). Вершина называется листом, если из неё исходит ровно одна тропинка, и она не является корнем.

Шуршик совершил  $q$  прогулок, каждая из которых проходила по нескольким правилам: сначала ёжик находил приятный путь, затем ко всем рёбрам пути добавлял  $+1$  к их протоптанности. Помогите Шуршику определить, сколько раз он побывает в каждой листовой вершине.

### Формат входных данных

Каждый тест состоит из нескольких наборов входных данных. Первая строка содержит одно целое число  $t$  ( $1 \leq t \leq 10^4$ ) — количество наборов входных данных. Далее следует описание наборов входных данных.

Первая строка содержит два целых числа  $n$  и  $q$  ( $2 \leq n \leq 2 \cdot 10^5$ ,  $0 \leq q \leq 10^{18}$ ) — количество вершин в дереве и количество прогулок Шуршика.

Следующие  $n - 1$  строк описывают рёбра дерева. Каждая строка содержит три целых числа  $u_i$ ,  $v_i$ ,  $w_i$  ( $1 \leq u_i, v_i \leq n$ ,  $0 \leq w_i \leq 10^{18}$ ) — номера вершин, соединённых тропинкой, и её начальная протоптанность.

Гарантируется, что сумма  $n$  по всем тестам не превышает  $2 \cdot 10^5$  и заданный граф является деревом, корнем которого является вершина 1.

### Формат выходных данных

Для каждого теста выведите информацию о посещениях листовых вершин.

Для каждой листовой вершины **в порядке возрастания** её номера выведите в отдельной строке два числа: номер листа и количество раз, которое Шуршик побывал в этом листе за все  $q$  прогулок.

### Система оценки

| Группа | Баллы | Дополнительные ограничения                            | Необходимые группы |
|--------|-------|-------------------------------------------------------|--------------------|
| 1      | 12    | $\sum n \cdot q \leq 2 \cdot 10^5$                    | —                  |
| 2      | 13    | $t = 1$ ; $n \leq 20$ ; $w_i = 0$                     | —                  |
| 3      | 20    | $u_i = \lfloor \frac{i+1}{2} \rfloor$ , $v_i = i + 1$ | —                  |
| 4      | 20    | $w_i = 0$ для всех рёбер                              | 2                  |
| 5      | 15    | $u_i = 1$ , $v_i = i + 1$                             | —                  |
| 6      | 20    | Нет дополнительных ограничений                        | 1, 2, 3, 4, 5      |



## Пример

| стандартный ввод | стандартный вывод |
|------------------|-------------------|
| 4                | 2 4               |
| 5 7              | 4 1               |
| 1 2 0            | 5 2               |
| 1 3 1            | 2 200             |
| 3 4 5            | 3 100             |
| 3 5 3            | 4 299             |
| 4 599            | 2 498089590       |
| 1 2 300          | 4 166029863       |
| 1 3 400          | 5 166029863       |
| 1 4 200          | 7 41507466        |
| 12 996179179     | 8 41507466        |
| 1 2 0            | 9 41507466        |
| 1 3 0            | 11 20753733       |
| 3 4 0            | 12 20753732       |
| 3 5 0            | 4 195             |
| 3 6 0            | 5 194             |
| 6 7 0            | 6 194             |
| 6 8 0            | 7 194             |
| 6 9 0            |                   |
| 6 10 0           |                   |
| 10 11 0          |                   |
| 10 12 0          |                   |
| 7 777            |                   |
| 1 2 0            |                   |
| 1 3 0            |                   |
| 2 4 0            |                   |
| 2 5 0            |                   |
| 3 6 0            |                   |
| 3 7 0            |                   |

## Замечание

$\lfloor \frac{a}{b} \rfloor$  означает округление вниз результата деления  $\frac{a}{b}$ . Так  $\lfloor \frac{9}{10} \rfloor = 0$  и  $\lfloor \frac{1}{10} \rfloor = 0$ .



## Задача D. Хомячья зрелищность

Ограничение по времени: 5 секунд

В знаменитой Хомячьей Академии проходит ежегодная олимпиада среди хомяков. В ряду сидят  $n$  хомяков в майках, на каждой майке написан личный рекорд хомяка по бегу в колесе — целое положительное число  $a_i$  (сколько оборотов в минуту может сделать хомяк). Каждое число-рекорд  $X$  можно разложить на особые навыки. Все навыки, которыми обладает хомяк — это числа, которые являются делителями  $X$ . Например, хомяк с рекордом 12 обладает навыками 1, 2, 3, 4, 6, 12. Чем больше уникальных навыков демонстрирует группа хомяков, тем зрелищнее выступление.

Для оценки судьи могут выбрать непрерывную команду хомяков — от  $L$ -го до  $R$ -го в ряду. Однако, чем больше хомяков в команде, тем сложнее их организовать и прокормить, и поэтому выступление становится менее зрелищным. Затраты на содержание команды равны её размеру  $(R - L + 1)$  и равны тому значению, на которое снижается зрелищность выступления соответствующей команды хомяков.

Помогите определить судьям, какая зрелищность выступлений будет максимальна. Итоговая зрелищность выступления определяется как  $F(L, R) = X - (R - L + 1)$ , где  $X$  — количество различных положительных целых чисел, которые являются особыми навыками хотя бы одного хомяка с рекордом  $a_i$ , такого что  $L \leq i \leq R$ ; а  $R - L + 1$  — затраты на содержание команды на отрезке от  $L$ -го до  $R$ -го хомяка в ряду.

### Формат входных данных

В первой строке дано одно целое число  $n$  ( $1 \leq n \leq 5 \cdot 10^5$ ) — длина ряда сидящих хомяков.

Во второй строке дано  $n$  целых чисел  $a_1, a_2, \dots, a_n$  ( $1 \leq a_i \leq 5 \cdot 10^6$ ) — номера, которые написаны на майках хомяков.

### Формат выходных данных

Выведите единственное число — максимальное значение зрелищности выступления хомяков  $F(L, R)$ .

### Система оценки

Тесты к этой задаче состоят из девяти групп. Баллы за каждую группу ставятся только при прохождении всех тестов группы, всех тестов необходимых для неё групп и тестов из условия.

| Группа | Баллы | Доп. ограничения      |                 | Необходимые группы |
|--------|-------|-----------------------|-----------------|--------------------|
|        |       | $n$                   | $a_i$           |                    |
| 1      | 16    | $n \leq 100$          | $a_i \leq 100$  |                    |
| 2      | 15    | $n \leq 10^3$         | $a_i \leq 10^3$ | 1                  |
| 3      | 11    | $n \leq 10^3$         | —               | 1, 2               |
| 4      | 10    | $n \leq 10^5$         | $a_i \leq 10^5$ | 1, 2               |
| 5      | 7     | $n \leq 10^4$         | —               | 1–3                |
| 6      | 8     | $n \leq 5 \cdot 10^4$ | —               | 1–3, 5             |
| 7      | 9     | $n \leq 10^5$         | —               | 1–6                |
| 8      | 9     | $n \leq 2 \cdot 10^5$ | —               | 1–7                |
| 9      | 15    | —                     | —               | 1–8                |



## Пример

| стандартный ввод   | стандартный вывод |
|--------------------|-------------------|
| 6<br>3 14 2 20 9 8 | 6                 |



## Задача E. Bracket Dance

Ограничение по времени: 1 секунда

Вы наконец добрались до сокровища! Конечно же это строка  $s$  длины  $n = 2^k$ , состоящая из скобок.

Скажем прямо — кот Персик<sup>†</sup> обожает скобочные последовательности и степени двойки!

Персик играет со строкой  $s$  — несколько раз (возможно, ноль) делит её на части и собирает обратно.

За один шаг происходит следующее:

Последовательность делится на блоки длины  $m = 2^x$  ( $x \geq 1$ ), каждый блок делится на две половины, половины внутри блока меняются местами.

Пример:

$x = 2, m = 4, s = (()) \Rightarrow |((\cdot))| \Rightarrow |)) \cdot ((| \Rightarrow (())$

$x = 2, m = 4, s = (((())) \Rightarrow |(((\cdot)))| \cdot () \Rightarrow |() \cdot (((\cdot)))| \Rightarrow ()((( )))$

$x = 1, m = 2, s = (((())) \Rightarrow |(\cdot(|(\cdot)))| \cdot () \Rightarrow |(\cdot(| \cdot (| \cdot )))| \cdot (| \Rightarrow ((()))$

Нам интересно, когда Персик наиграется (то есть ему надоест менять исходную строку) — для этого помогите нам узнать (в зависимости от типа запроса):

- Сколько существует различных вариантов получить правильную скобочную последовательность<sup>‡</sup> (ПСП) из исходной строки  $s$ , пока Персик с ней играет. Два варианта считаются различными, если для некоторого  $i$  символ, стоящий изначально на позиции  $i$  переместился на разные позиции в этих двух вариантах.
- Сколько существует различных ПСП, которые можно получить из исходной строки. Две ПСП  $a, b$  считаются различными, если существует индекс  $i$  такой, что  $a_i \neq b_i$ .

<sup>†</sup> Персик — акула в мире программирования.

<sup>‡</sup> Скобочная последовательность называется правильной, если ее можно превратить в корректное математическое выражение, добавляя только символы  $+$  и  $1$ . Например, последовательности  $((()))()$ ,  $()$ ,  $((()))()$  и пустая строка являются правильными, а  $)()$ ,  $((()$  и  $((()))()$  — нет.

### Формат входных данных

В первой строке вводится одно число  $T$  — количество тестовых примеров ( $1 \leq T \leq 10\,000$ ).

Во второй строке вводится одно число  $Q$  — тип запроса (общий для всех тестовых примеров) ( $1 \leq Q \leq 2$ ).

В следующих  $T$  строках вводится по одной скобочной последовательности. ( $2 \leq |s_i| \leq 2^{22}$ ).

Суммарная длина всех вводимых последовательностей не превосходит  $2^{22}$  ( $\sum_{i=1}^T |s_i| \leq 2^{22}$ ).

### Формат выходных данных

Для каждой строки выведите требуемое число:

При  $Q = 1$  — количество различных перестановок, т.ч. итоговая строка ПСП.

При  $Q = 2$  — количество различных ПСП, которые можно получить такими действиями.

### Система оценки

| Группа | Баллы | Дополнительные ограничения              | Необходимые группы |
|--------|-------|-----------------------------------------|--------------------|
| 1      | 18    | $q = 1, \sum_{i=1}^T  s_i  \leq 2^8$    | —                  |
| 2      | 26    | $q = 1, \sum_{i=1}^T  s_i  \leq 2^{14}$ | 1                  |
| 3      | 26    | $q = 1, \sum_{i=1}^T  s_i  \leq 2^{22}$ | 1, 2               |
| 4      | 13    | $q = 2, \sum_{i=1}^T  s_i  \leq 2^{12}$ | —                  |
| 5      | 17    | $q = 2, \sum_{i=1}^T  s_i  \leq 2^{17}$ | 4                  |



## Примеры

| стандартный ввод            | стандартный вывод |
|-----------------------------|-------------------|
| 2<br>1<br>))((<br>((( )))() | 2<br>2            |
| 2<br>2<br>))((<br>((( )))() | 1<br>2            |

## Замечание

Если взять строку  $s = ({}^1(2)^3)^4$  (после символа указана его изначальная позиция), то она сама является ПСП, а также после одного шага при  $x = 1, m = 2$  получим строку  $({}^2(1)^4)^3$ . Итого есть два варианта.

На иллюстрации изображено применение одного шага  $x = 2, m = 4$  к строке  $((()))()$ .

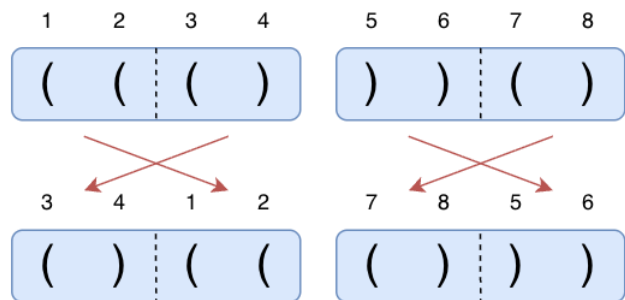


Иллюстрация к примеру