

Условия задач 9-11 класс

Войдите в тестирующую систему.

Муниципальный этап Всероссийской олимпиады Калужская область. Программирование. 2025-2026 (9-11 класс)

<https://official.contest.yandex.ru/contest/84529>

Нажмите на ссылку «Войти» в верхнем правом углу страницы (или на ссылку «Авторизоваться»).

Введите логин и пароль, указанные на листе, который выдан организаторами олимпиады.

После входа в систему будут доступны закладки: «Задачи», «Посылки», «Сообщения».

Решением для задач является **консольное приложение!**

Данные можно считывать с консоли или из текстового файла `input.txt`. Результат можно выводить на консоль или в текстовый файл `output.txt`.

Каждая строка во входных данных завершается символом конца строки. Последняя строка во входном файле пустая.

Система оценивания

Если не пройдены **все** тесты из условия, то баллы за задачу **не начисляются**.

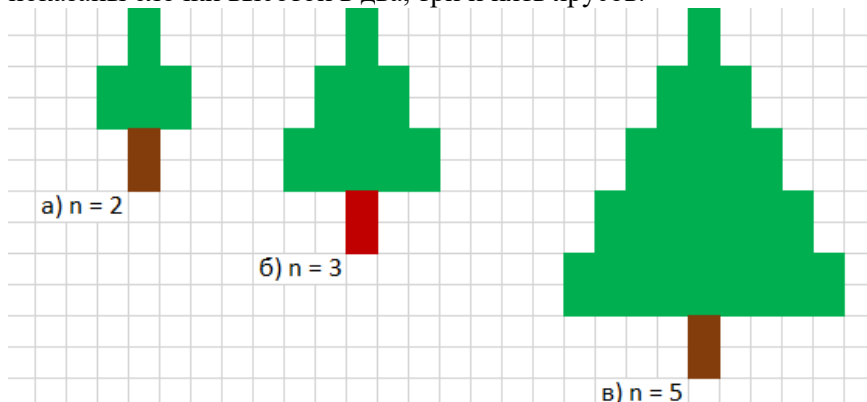
Баллы начисляются за каждый пройденный тест (кроме тестов из условия). Если все тесты для данной подзадачи пройдены, то начисляется указанное для этой подзадачи количество баллов. Если тесты пройдены не все, то начисляется меньшее количество баллов. Решение получает максимальное количество баллов в случае, когда **все** тесты будут пройдены.

Максимальное время тестирования всех задач на любом тесте – 1 секунда (если в условии не оговорено иное).

1. «Ёлочка» (100 баллов)

Максимальное время тестирования на одном тесте – 0,25 с

Руслан решил сложить ёлочку из мозаики. Для создания каждого яруса он выкладывает нечетное количество элементов мозаики в два ряда. В первом ярусе в каждом ряду используется один элемент мозаики. В каждом следующем ярусе в ряду на два элемента мозаики больше, чем в предыдущем. Во втором ярусе в каждом ряду используется три элемента мозаики и т.д. У ёлочки есть еще ножка из двух элементов мозаики. На рисунке показаны ёлочки высотой в два, три и пять ярусов.



Руслана заинтересовал вопрос: как определить, сколько всего элементов мозаики понадобится, чтобы собрать ёлочку из N ярусов?

Входные данные:

В первой и единственной строке записано натуральное число N – количество ярусов в ёлочке ($1 \leq N \leq 2 \cdot 10^9$).

Выходные данные:

Одно целое число – количество элементов мозаики, необходимых для создания ёлочки.

Пример:

Входные данные	Выходные данные
3	20

Примечание

Обратите внимание, что в данной задаче необходимо использовать 64-битный тип данных, например, `longlong` в C++, `int64` в Free Pascal, `long` в Java.

Описание подзадач:

Подзадача	Баллы	Дополнительные ограничения
1	70	$1 \leq N \leq 10^7$
2	30	Ограничения из условия

2. «Амулет» (100 баллов)

В некоторой волшебной стране настали неблагоприятные времена. Чтобы пережить непростые времена было решено обратиться к магам. Они предложили изготовить защитный амулет и нанести на него подстроку магического заклинания из волшебной книги. Заклинание состоит из латинских букв. Каждый символ заклинания обладает магической силой. Чем больше символов заклинания будет на амулете, тем сильнее он будет. Однако, существует жесткое правило, если хотя бы один из символов X, Y или Z будет встречаться более трех раз на амулете, то амулет потеряет свою силу. Необходимо изготовить амулет наибольшей магической силы.

Входные данные:

В первой и единственной строке записаны латинские буквы заклинания из волшебной книги. Длина строки не превышает 10^6 символов.

Выходные данные:

В первой строке необходимо записать длину подстроки заклинания, которое будет нанесено на амулет.

Во второй строке необходимо записать сами символы заклинания, которые будут нанесены на амулет. Если возможных вариантов подстроки несколько, то нужно вывести первую, которая встречается в заклинании.

Пример:

Входные данные	Выходные данные
XXABYCYZCDYXX	11 XXABYCYZCDYX

Описание подзадач:

Подзадача	Баллы	Дополнительные ограничения
1	30	Длина исходной строки не превышает 1000
2	40	Длина исходной строки не превышает 10000
3	30	Ограничения из условия

3. «Машина времени» (100 баллов)

Представим, что машина времени все-таки изобретена. Но пока невозможно дать доступ всем желающим. Изобретатели решили защитить свое изобретение следующим образом: чтобы выйти из машины после перемещения в прошлое или будущее нужно открыть кодовый замок, принцип работы которого связан с простыми числами. Простым числом называется число, которое имеет ровно два натуральных делителя.

Путешественнику во времени, чтобы покинуть машину времени, когда он переместился в прошлое или будущее, нужно ввести количество простых чисел, находящихся на некотором отрезке. Каждый раз предлагаются разные отрезки. Если достаточно быстро не ввести правильный ответ или ввести его неправильно, то кодовый замок блокируется и путешественника возвращают назад в свое время.

Исследователю очень нужно побывать в N разных моментах времени. Он просит, чтобы вы помогли ему быстро получить количество простых чисел на N отрезках.

Входные данные:

В первой строке записано натуральное число N – количество моментов времени, которых нужно посетить исследователю ($1 \leq N \leq 10^5$).

В каждой из следующих N строк записаны через пробел по два целых числа a_i и b_i – границы отрезка, на котором нужно быстро определить количество простых чисел, чтобы можно было покинуть машину времени в этот момент ($1 \leq a_i < b_i \leq 10^6$).

Выходные данные:

N целых чисел – количество простых чисел на указанных отрезках. Каждое число выводится с новой строки.

Пример:

Входные данные	Выходные данные
3	4
1 10	3
5 11	2
4 9	

Описание подзадач:

Подзадача	Баллы	Дополнительные ограничения
1	30	$1 \leq N \leq 100, 1 \leq a_i < b_i \leq 1000$
2	40	$1 \leq N \leq 1000, 1 \leq a_i < b_i \leq 1000$
3	20	$1 \leq N \leq 1000, 1 \leq a_i < b_i \leq 10000$
4	10	Ограничения из условия

4. «Горилла и строковая игра» (100 баллов)

Однажды огромный самец гориллы заскучал и решил сыграть со своим другом бегемотом в строковую игру. Даны две строки одинаковой длины s и t одинаковой длины. Игроки ходят по очереди, начинает игру горилла. В свой ход игрок **обязан** выполнить одно из двух действий:

1. Удалить непустой префикс строки s , состоящий из **одинаковых** символов; удалить столько же символов с конца строки t ;
2. Удалить непустой префикс строки t , состоящий из **одинаковых** символов; удалить столько же символов с конца строки s .

Игрок, не способный сделать ход, проигрывает. Горилла и бегемот сыграют m партий. Определите победителя в каждой партии, если оба игрока играют оптимально.

Входные данные:

В первой строке записано целое число m – количество партий ($1 \leq m \leq 100$).

В следующих m строках описаны партии.

В первой строке каждой партии дана строка s , состоящая из символов `a`, `b`, `c` ($1 \leq |s| \leq 50$). Длина строки s обозначается как $|s|$.

Во второй строке каждой партии дана строка t , состоящая из символов `a`, `b`, `c` ($|t| = |s|$). Длина строки t обозначается как $|t|$.

Гарантируется, что длины строк s и t равны.

Выходные данные:

Выведите для каждой партии в новой строке «Gorilla», если победит горилла, в противном случае выведите «Begemot».

Примеры:

Входные данные	Выходные данные
4	Gorilla
cca	Gorilla
bab	Begemot
bbacaba	Begemot
aabbbbc	
accacca	
bcccaa	
abbbbc	
cbbbbbb	

Пояснение к примеру

Разберём первую партию теста из условия.

Если горилла удаляет «cc» из s , бегемот удаляет «d» из s , горилла проигрывает.

Если горилла удаляет «b» из t , бегемот удаляет «cc» из s , горилла проигрывает.

Если горилла удаляет «c» из s , бегемот удаляет «c» из s или «b» из t , затем горилла удаляет последнюю букву и выигрывает.

Оба игрока играют оптимально, поэтому горилла делает выигрышный ход и побеждает в партии

Описание подзадач

Подзадача	Баллы	Дополнительные ограничения
1	24	Длины строк s и t не превышают 8
2	18	$s_i \neq s_{i+1}, t_i \neq t_{i+1}$
3	18	$m \leq 4$
4	40	Ограничения из условия

5. «Последняя олимпиада Саши» (100 баллов)

Максимальное время тестирования на одном тесте – 2 с

Саша всю ночь готовился к городской олимпиаде по литературе. По сугробам, сонный, но уверенный в себе, он добрёл до точки проведения. Дверь в пустую аудиторию захлопнулась, за окном застыла крошечная тьма, тиканье часов остановилось. Зловещая чёрная фигура нагнулась над Сашей и вместо стихотворения для анализа вручила ему массив из n целых чисел $a_1, a_2, \dots, a_n$.

Саша должен ответить на m запросов. Запросы бывают двух видов:

«=» – заменить **все** элементы подотрезка $a_l, a_{l+1}, \dots, a_r$ на

$$1 + \max(a_l, a_{l+1}, \dots, a_r) - \min(a_l, a_{l+1}, \dots, a_r);$$

«?» – посчитать количество простых делителей, входящих в разложение числа $a_l \cdot a_{l+1} \cdot \dots \cdot a_r$ в **нечётной** степени.

Саша не был готов к такому повороту событий. Помогите ему справиться с заданием.

Входные данные

В первой строке даны целые числа n, m – количество элементов в массиве и количество запросов соответственно ($1 \leq n, m \leq 200\,000$).

Во второй строке даны n целых чисел $a_1, a_2, \dots, a_n$ – элементы массива.

В следующих m строках описаны запросы. Запросы бывают двух видов:

- $= l\ r$ – запрос первого вида ($1 \leq l \leq r \leq n$);
- $? l\ r$ – запрос второго вида ($1 \leq l \leq r \leq n$).

Выходные данные

Выведите ответ на каждый запрос типа «?» в отдельной строке.

Примеры:

Входные данные	Выходные данные
7 6	1
4 3 1 256 18 22 100	2
? 2 3	1
? 1 5	0
= 4 6	
? 4 4	
= 1 5	
? 1 7	

Пояснение к примеру

Рассмотрим запросы в тесте из условия.

1. $3 \cdot 1 = 3 = 3^1$, поэтому ответ 1;
2. $4 \cdot 3 \cdot 1 \cdot 256 \cdot 18 = 55296 = 2^{11} \cdot 3^3$, поэтому ответ 2;
3. Массив превращается в 4, 3, 1, 239, 239, 239, 100;
4. $239 = 239^1$, поэтому ответ 1;
5. Массив превращается в 239, 239, 239, 239, 239, 239, 100;
6. $239^6 \cdot 100 = 2^2 \cdot 5^2 \cdot 239^6$, поэтому ответ 0.

Примечание

Обратите внимание, что в данной задаче необходимо использовать 64-битный тип данных, например `longlong` в C++, `int64` в Free Pascal, `long` в Java.

Если вы сдаете решение на **Python**, то

- сдавайте решение на PyPy;
- вставьте в начало файла `from sys import stdin`;
- используйте `stdin.readline()` вместо `input()`;
- минимизируйте количество вызовов `print`.

Если вы сдаете решение на **Pascal**, то сдавайте решение на Free Pascal.

Если вы сдаете решение на C++, то

- используйте `\n` вместо `std::endl`
- В самом начале функции `main` пропишите строчку
`std::ios_base::sync_with_stdio(false);`
`std::cin.tie(nullptr);`
Это ускорит ввод/вывод чисел.

Описание подзадач

Подзадача	Баллы	Дополнительные ограничения
1	16	$n, m \leq 500$
2	8	$n, m \leq 10\,000$
3	14	только запросы вида «?»
4	10	$a_i \leq 2$
5	18	$a_i \leq 30$
6	14	$r - l \leq 50$ в запросах вида «=»
7	20	Ограничения из условия