

Задача А. К-интересные подотрезки

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

В древнем королевстве Бурляндия мудрый математик Вася открыл таинственное свойство некоторых массивов. Согласно древним свиткам, k -интересные массивы содержат ключ к поиску скрытых сокровищ по всему королевству.

Вася определяет массив a длины n как k -интересный, если существует такой индекс i ($1 \leq i < n$), что:

$$(a_1 + a_2 + \dots + a_i) \cdot a_{i+1} \cdot a_{i+2} \cdot \dots \cdot a_n = k$$

Иными словами, сумма первых i элементов, умноженная на произведение оставшихся элементов, равна k .

Король Артур, очарованный этими массивами, владеет картой, представленной в виде массива длины n . Он подозревает, что некоторые части этой карты могут привести к различным сокровищам, каждое из которых связано с определенным значением k .

Королевские советники подготовили q запросов для анализа этой карты. Всего есть два типа запросов. Каждый запрос первого типа задаётся тремя числами l , r и k . Ваша задача — проверить, правда ли, что подмассив массива a с l -го по r -й элементы является k -интересным. Кроме того, советники иногда обновляют карту новой информацией. Эти обновления представлены как запросы второго типа. Каждый запрос второго типа задан двумя числами i и x , которые обозначают, что теперь i -й элемент массива a равен x .

Помогите королю Артуру найти сокровища до того, как их обнаружат другие королевства!

Формат входных данных

Первая строка содержит два целых числа n и q ($2 \leq n \leq 2 \cdot 10^5, 1 \leq q \leq 2 \cdot 10^5$) — длина массива a и количество запросов.

Вторая строка содержит n целых чисел $a_1, a_2, \dots, a_n$ ($1 \leq a_i \leq 10^9$) — элементы массива a .

Каждая из следующих q строк описывает запрос в одном из следующих форматов:

- $1 \ l \ r \ k$ — нужно проверить, является ли подмассив $[l; r]$ массива a k -интересным ($1 \leq l < r \leq n, 1 \leq k \leq 10^{18}$).
- $2 \ i \ x$ — нужно присвоить a_i значение x ($1 \leq i \leq n, 1 \leq x \leq 10^9$).

Формат выходных данных

Для каждого запроса первого типа выведите «YES», если указанный подмассив является k -интересным, или «NO» в противном случае.

Система оценки

Баллы за каждую подзадачу начисляются только в случае, если все тесты для этой и необходимых подзадач успешно пройдены.

Подзадача	Баллы	Доп. ограничения	Необх. подзадачи
1	12	$n, q \leq 200$	
2	9	$n \leq 200$, запросы только 1-го типа	
3	14	$n, q \leq 3000$, запросы только 1-го типа	
4	7	$n, q \leq 3000$	1,3
5	15	В любой момент a_i чётно, запросы только 1-го типа	
6	12	В любой момент a_i чётно	5
7	20	Запросы только 1-го типа	2,3,5
8	11	—	1,2,3,4,5,6,7

Примеры

стандартный ввод	стандартный вывод
5 4 5 2 7 10 2 1 2 4 90 2 4 11 1 2 4 90 1 1 5 308	YES NO YES
7 6 4 6 4 2 9 10 7 2 7 3 2 4 8 1 3 4 32 2 7 7 2 3 10 1 4 6 720	YES YES

Замечание

Разберем первый пример из условия. В первом запросе нас просят проверить, является ли подмассив $[2, 7, 10]$ 90-интересным. Он является таковым, так как $(2 + 7) \cdot 10 = 90$.

После второго запроса массив выглядит следующим образом: $[5, 2, 7, 11, 2]$.

В третьем запросе нас просят проверить, является ли подмассив $[2, 7, 11]$ 90-интересным. Он таковым не является, так как мы не можем выбрать такое i , чтобы произведение суммы первых i элементов и остальных элементов было равно 90.

В четвертом запросе нас просят проверить, является ли массив целиком 308-интересным. Он является таковым, так как $(5 + 2 + 7) \cdot 11 \cdot 2 = 308$.

Задача В. Гонки

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Вася приехал в Монако на гонки «Формулы 1». Всего гонка будет идти n дней. Его любимая команда — «Педальки вниз». В каждый из этих дней Вася знает скорость, с которой будет ехать болид этой команды: в i -й день скорость составит v_i .

Так как Вася хотел не только посетить гонки, но и сходить на пляж и хорошенько отдохнуть, он решил, что приедет хотя бы на k дней. При этом покупать слишком много билетов он не хочет, поэтому он будет отдыхать непрерывный отрезок из нескольких дней. Но чтобы его отдых оказался максимально хорошим, Вася хочет, чтобы средняя скорость болида его любимой команды за все дни, которые он проведет в Монако, была как можно больше.

Более формально, вам нужно найти подотрезок длины хотя бы k , чтобы среднее арифметическое значений этого подотрезка было как можно больше.

Формат входных данных

В первой строке вводятся два целых числа n и k — продолжительность «Формулы 1» и минимальное количество дней, которое Вася хочет отдыхать ($1 \leq k \leq n \leq 10^5$, $n \cdot k \leq 10^6$).

Во второй строке через пробел вводятся n целых чисел $v_1, v_2, \dots, v_n$ — скорости болида в каждый из дней ($0 \leq v_i \leq 10^9$).

Формат выходных данных

Выведите два целых числа l и r ($1 \leq l \leq r \leq n$) — начало и конец искомого отрезка. Если оптимальных решений несколько, выведите любое из них.

Система оценки

Баллы за каждую подзадачу начисляются только в случае, если все тесты для этой и необходимых подзадач успешно пройдены.

Подзадача	Баллы	Доп. ограничения	Необх. подзадачи
1	21	$n \leq 100$	
2	24	$n \leq 1000$	1
3	55	—	1,2

Примеры

стандартный ввод	стандартный вывод
4 2 1 2 3 4	3 4
10 2 3 5 2 7 2 9 3 8 2 7	6 8

Замечание

В первом примере Вася будет отдыхать 2 последних дня, средняя скорость болида в эти дни составит 3.5.

Во втором примере оптимальным вариантом будет отдыхать 3 дня — с 6 по 8 включительно. Во всех вариантах отдыха в течение двух дней средняя скорость будет ниже.

Задача С. Студент и электрички

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

В Берляндии n городов, соединённых последовательно железной дорогой. В каждом i -м городе есть станция, на которой есть только одна платформа высоты h_i . Между городами курсирует m маршрутов электричек разных классов. Электричка j -го маршрута имеет класс w_j и следует в обоих направлениях между городами l_j и r_j с остановками во всех городах k , таких, что $l_j \leq k \leq r_j$ и $h_k \leq t_j$, где t_j — высота пола электрички (обратите внимание, что электричка может не останавливаться в городах l_j и r_j).

Чтобы путешествовать на электричках, Вася за x монет может купить проездной, позволяющий ему пользоваться электричками всех классов от 1 до x .

Вася выбирает, в какой университет ему лучше поступить. Для каждого из q университетов Берляндии он знает, в каком городе расположен учебный корпус, и в каком — общежитие, поэтому просит вас рассчитать минимальную необходимую стоимость проездного, чтобы перемещаться между этими городами.

Формат входных данных

Первая строка содержит три натуральных числа n , m и q ($1 \leq n, m, q \leq 10^5$) — количество городов, маршрутов электричек и университетов.

Вторая строка содержит n натуральных чисел h_i ($1 \leq h_i \leq 10^5$) — высоты платформ.

Каждая из следующих m строк содержит 4 натуральных числа l_j , r_j , w_j и t_j ($1 \leq l_j \leq r_j \leq n$, $1 \leq w_j, t_j \leq 10^5$) — первую и последнюю станции маршрута, класс и высоту пола электрички.

Последние q строк содержат по два натуральных числа c_k и d_k ($1 \leq c_k, d_k \leq n$) — города, в которых расположены учебный корпус и общежитие университета.

Формат выходных данных

Выведите q целых чисел — для каждого университета выведите минимальную стоимость проездного, необходимого, чтобы перемещаться между городами c_k и d_k . Если не существует способа доехать из города c_k в город d_k , выведите на соответствующей строке вместо ответа -1 .

Система оценки

Баллы за каждую подзадачу начисляются только в случае, если все тесты для этой и необходимых подзадач успешно пройдены.

Подзадача	Баллы	Доп. ограничения	Необх. подзадачи
1	15	$n, m, q \leq 100$	
2	10	$h_i = 1, n, m, q \leq 1000$	
3	10	$n, m, q \leq 1000$	1,2
4	5	$h_i = 1, n, q \leq 1000$	2
5	15	$h_i = 1$	2,4
6	45	—	1,2,3,4,5

Примеры

стандартный ввод	стандартный вывод
5 3 4 4 3 1 3 5 3 5 4 5 3 5 2 3 1 3 3 5 1 5 4 3 3 1 4 1	4 2 3 3
5 2 3 4 4 2 1 4 1 3 3 5 4 4 4 5 4 3 1 3 1 4	-1 3 -1

Замечание

В первом примере Вася может доехать из города 1 до города 5 с проездным стоимостью 4 — сначала доехать на 3-й электричке (уровня 3) до города 3, затем доехать до города 5 на 1-й электричке (уровня 4).

В этом же примере Вася может доехать из города 4 в город 3 на 2-й электричке (уровня 2) с проездным стоимостью 2, и из города 3 в город 1 на 3-й электричке (уровня 3) с проездным стоимостью 3. Поэтому он также может доехать из города 4 в город 1 с проездным стоимостью 3.

Во втором примере Вася не сможет доехать из города 4 до городов 1 и 3. При этом он может доехать из города 1 в город 3 на 1-й электричке (уровня 3).

Задача D. Бег с препятствиями

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Спортсмен Миша попал на интересный бег с препятствиями. Обычно препятствия расположены на беговой полосе, и их нужно перепрыгивать. Но на этот раз беговых полос целых две, а через препятствия перепрыгивать запрещено, а можно только перестраиваться между двумя полосами.

Забег происходит по следующим правилам:

- Миша стартует с точки 0 на произвольной полосе.
- Пусть Миша находится в точке x , тогда он может попасть в точку $x + 1$ той же полосы, если на этой точке нет препятствия.
- Пусть Миша находится в точке x , тогда он может попасть в точку $x + 1$ другой полосы, если на этой точке нет препятствия.
- Забег завершается либо когда Миша достигает точки n , либо если не может попасть в следующую клетку на обеих полосах.

Узнайте, как далеко Миша сможет пробежать, если:

- препятствия на первой полосе расположены в точках $a - k, a - k + 1, a - k + 2, \dots, a - 1$, a и повторяются через a на всей полосе, то есть следующие препятствия после a находятся в точках $2 \cdot a - k, \dots, 2 \cdot a$ и так далее.
- препятствия на второй полосе расположены в точках $b - k, b - k + 1, b - k + 2, \dots, b - 1, b$ и повторяются через b на всей полосе, то есть следующие препятствия после b находятся в точках $2 \cdot b - k, \dots, 2 \cdot b$ и так далее.

Формат входных данных

В единственной строке задается 4 числа k, a, b, n ($0 \leq k < a, b, n \leq 10^9$).

Формат выходных данных

Выведите одно число — до какой точки сможет пробежать Миша.

Система оценки

Баллы за каждую подзадачу начисляются только в случае, если все тесты для этой и необходимых подзадач успешно пройдены.

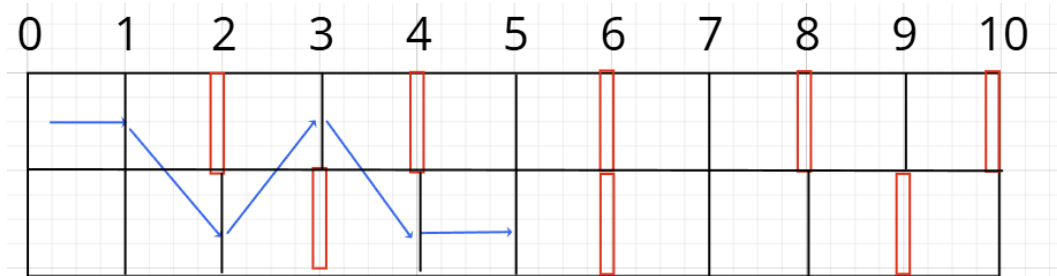
Подзадача	Баллы	Доп. ограничения	Необх. подзадачи
1	21	$k = 0$	
2	17	$a = 2$	
3	24	$n \leq 10^5$	
4	38	—	1,2,3

Примеры

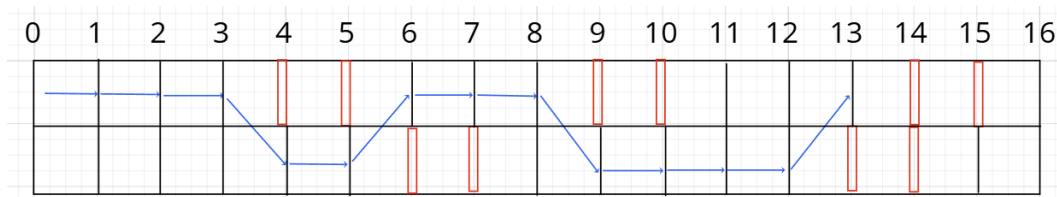
стандартный ввод	стандартный вывод
0 2 3 10	5
1 5 7 16	13
2 6 10 9	9

Замечание

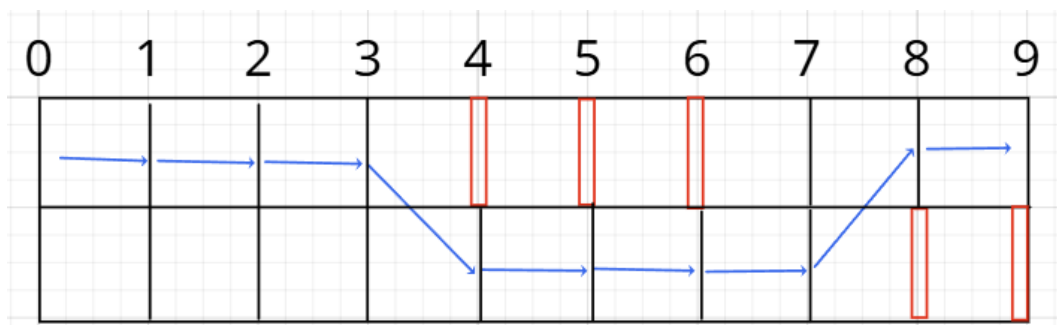
Пример 1:



Пример 2:



Пример 3:



Задача Е. Просто песня

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Недавно Эдуард услышал новую песню. Эта песня ему очень понравилась, однако, как это часто бывает с новыми песнями, слова в ней разобрать он не смог. Это неприятное событие ничуть не смутило Эдуарда — у него и так богатый словарный запас. Почему бы не составить из знакомых слов свой вариант песни? Чтобы напевать услышанную мелодию было удобнее, Эдуард решил составить из этих слов такую последовательность, что позиции букв под ударением останутся теми же.

Слово — это последовательность букв, **ровно на одной** из которых стоит ударение. Эдуард знает n различных слов, а услышанная песня состоит из m слов. Слова в песне могут повторяться, а также в песне могут встречаться слова, которые Эдуард не знает (он просто записал то, что услышал), однако каждое такое слово все равно содержит ровно одно ударение.

Последовательность из нескольких слов, которые знает Эдуард, назовем *удобной*, если **общее количество букв** в ней совпадает с количеством букв в услышанной песне, при этом **позиции букв под ударением** в последовательности и в песне совпадают. Отсюда, в частности, следует, что такая последовательность должна состоять ровно из m слов (поскольку в каждом слове ровно одна ударная буква). При этом длины соответствующих слов в последовательности и в песне могут различаться (см. примеры). Разрешается, чтобы последовательность содержала одинаковые слова.

Посчитайте, сколько различных удобных последовательностей может составить Эдуард. Две последовательности считаются различными, если они отличаются хотя бы в одном слове.

Формат входных данных

Первая строка содержит натуральное число n — количество слов, которые знает Эдуард ($1 \leq n \leq 10^6$).

Каждая из следующих n строк содержит по одному слову. Все n слов различны (слова, отличающиеся только позицией ударения, считаются различными).

Следующая строка содержит натуральное число m — количество слов в песне ($1 \leq m \leq 10^6$).

В последней строке заданы слова песни, каждые два соседних слова разделены одним пробелом.

Длина каждого слова как в списке Эдуарда, так и в песне не превосходит $2 \cdot 10^5$ символов. В каждом из слов есть ровно одна заглавная латинская буква — она обозначает позицию, на которой находится ударение. Все остальные символы являются строчными латинскими буквами.

Суммарная длина слов в списке Эдуарда — s_n — не превосходит 10^6 . Суммарная длина слов в песне (без учета пробелов) — s_m — не превосходит 10^6 .

Формат выходных данных

Выведите одно число — количество различных удобных последовательностей. Поскольку ответ может быть очень большим, выведите лишь его остаток от деления на число 998244353.

Система оценки

Баллы за каждую подзадачу начисляются только в случае, если все тесты для этой и необходимых подзадач успешно пройдены.

Подзадача	Баллы	Доп. ограничения	Необх. подзадачи
1	26	$s_n \leq 100, s_m \leq 100$	
2	16	$s_n \leq 100, s_m \leq 10^5$	1
3	26	$s_n \leq 10^5, s_m \leq 10^5$	1, 2
4	32	—	1,2,3

Примеры

стандартный ввод	стандартный вывод
4 yoU Are All wInners 4 yoU Are All wInners	4
5 delIcious A daY t0 Is 4 yoU Are All wInners	2

Замечание

В первом примере можно составить следующие удобные последовательности:

yoU Are All wInners

yoU All All wInners

yoU All Are wInners

yoU Are Are wInners

Во втором примере подходят следующие последовательности:

daY Is t0 delIcious

daY A daY delIcious

Задача F. Достойное продолжение

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

У Дианы есть набор из n различных целых чисел a_i . Каждое из чисел не менее 10, то есть состоит хотя бы из двух цифр.

Диана любит выписывать пары различных чисел подряд — одно за другим. Если число a_j записано после числа a_i , то Диана называет это *продолжением*. При этом Диана является экспертом в области эстетики! Наиболее красивым продолжением, по её мнению, является ситуация, когда последняя цифра числа a_i совпадает с первой цифрой числа a_j . Такой вариант Диана называет *достойным продолжением*, а саму пару чисел (a_i, a_j) — *достойной парой*. Например, пара чисел (19, 98) — достойная, а пара чисел (2025, 12) — нет.

Зная числа, которые есть у Дианы, посчитайте, сколько среди них есть достойных пар. Обратите внимание, что число не может быть само с собой в паре. Кроме того, пары (a_i, a_j) и (a_j, a_i) считаются различными. В частности, одна из этих пар может быть достойной, а другая — нет. Числа Дианы не содержат ведущих нулей (не начинаются с цифры 0), поэтому в любой достойной паре первое число не может заканчиваться на 0.

Формат входных данных

Первая строка содержит число n — количество чисел Дианы ($1 \leq n \leq 2 \cdot 10^5$).

Вторая строка содержит n различных чисел a_i — числа Дианы ($10 \leq a_i \leq 10^9$).

Формат выходных данных

Выведите одно число — количество достойных пар.

Система оценки

Баллы за каждую подзадачу начисляются только в случае, если все тесты для этой и необходимых подзадач успешно пройдены.

Подзадача	Баллы	Доп. ограничения	Необх. подзадачи
1	32	$a_i \leq 99$	
2	33	$n \leq 1000$	
3	35	—	1,2

Пример

стандартный ввод	стандартный вывод
5 19 98 2025 12 99	4

Замечание

В примере достойными являются следующие пары чисел: (19, 98), (19, 99), (99, 98), (12, 2025).

Задача G. Максимальный XOR

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Миша любит числа и часто пытается их максимизировать. В этот раз у него есть n чисел, он попросил своих сокомандников назвать еще два числа x и y .

Теперь Миша хочет максимизировать XOR (см. замечание) чисел во всем массиве. Для этого он хочет выполнить следующую операцию ровно один раз — взять два любых числа в массиве и к одному добавить x , а к другому y . Нельзя добавлять и x , и y к одному и тому же элементу массива.

Помогите ему найти максимальный XOR, который можно получить.

Формат входных данных

В первой строке дано три числа n, x, y ($2 \leq n \leq 2 \cdot 10^5, 0 \leq x, y \leq 10^9$).

В следующей строке задано n чисел a_i ($0 \leq a_i \leq 10^9$) — элементы массива.

Формат выходных данных

Выведите одно число — максимально возможный XOR.

Система оценки

Баллы за каждую подзадачу начисляются только в случае, если все тесты для этой и необходимых подзадач успешно пройдены.

Подзадача	Баллы	Доп. ограничения	Необх. подзадачи
1	13	$n \leq 100$	
2	19	$n \leq 10^3$	1
3	14	$x = 0$	
4	17	$a_i \leq 10^3$	
5	37	—	1,2,3,4

Пример

стандартный ввод	стандартный вывод
5 2 3 1 2 3 4 5	14

Замечание

Операцией XOR (или побитовое исключающее ИЛИ) называется операция над двумя целыми числами, при которой i -й разряд результата в двоичной системе счисления будет равен 1 тогда и только тогда, когда **ровно у одного** из двух целых чисел в i -м разряде в двоичной системе счисления стоит 1. XOR множества чисел получается путем последовательной замены пары чисел множества на XOR этих чисел, пока во множестве не останется одно число. Это число и будет результатом операции XOR над всем множеством.

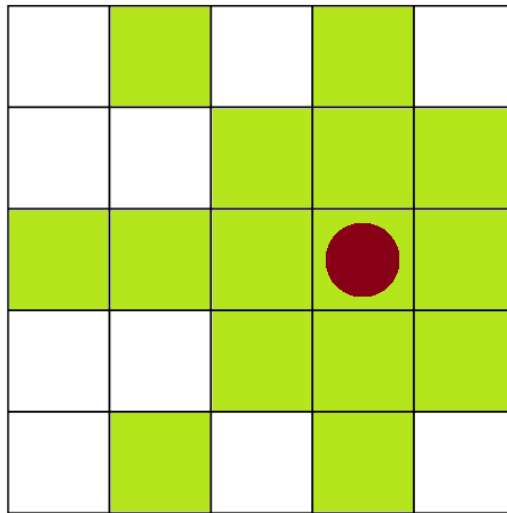
В примере можно к первому элементу прибавить 2, а к последнему — 3, тогда массив $[1, 2, 3, 4, 5]$ превратится в $[3, 2, 3, 4, 8]$ и XOR всего массива будет равен 14.

Задача Н. Фёдор и ферзи

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Фёдор любит играть в шахматы! При этом он не ограничивается привычными досками 8×8 , а использует доски $n \times n$. Для простоты пронумеруем горизонтали доски от 1 до n сверху-вниз, а вертикали — от 1 до n слева-направо.

У Фёдора есть k ферзей, которые ходят по обычным шахматным правилам — на любое количество клеток по горизонтали, вертикали и диагонали. Если поставить ферзя на шахматную доску, то клетки, на которые он может сделать ход, будут находиться *под боем*. Например:



зеленые клетки находятся под боем ферзя на позиции (3, 4)

Если на доске стоят несколько ферзей, то клетки, на которых стоят ферзи, а также клетки, которые находятся под боем хотя бы одного ферзя, будем называть *защищенными*.

Фёдор не любит гостей! Поэтому он хочет поставить на доску своих k ферзей так, чтобы защищенных клеток было как можно больше — тогда другие игроки не смогут поставить свои фигуры на эти клетки.

Помогите Фёдору — найдите оптимальную расстановку ферзей.

Формат входных данных

Первая строка содержит два натуральных числа — n и k — размер доски и количество ферзей ($1 \leq n \leq 16$, $1 \leq k \leq n$, $2 \cdot k \leq n + 1$).

Формат выходных данных

В первой строке выведите число m — максимальное количество клеток, которое может быть защищено ферзями.

В каждой из следующих k строк выведите два числа x_i , y_i — позицию очередного ферзя ($1 \leq x_i, y_i \leq n$). Позиции всех ферзей должны быть различны. Если существует несколько оптимальных расстановок, выведите любую из них.

Система оценки

Баллы за подзадачи 1 – 5 начисляются только в случае, если **все тесты** для этой и необходимых подзадач успешно пройдены.

Баллы за подзадачи 6 – 9 начисляются за **каждый пройденный тест**, если все тесты для необходимых подзадач успешно пройдены.

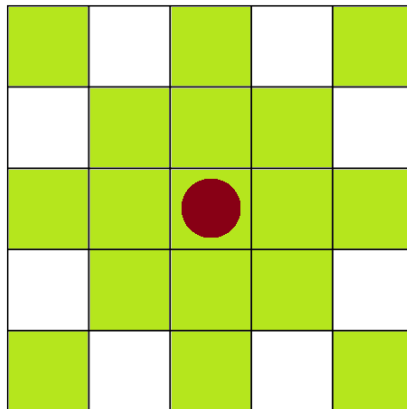
Подзадача	Баллы	Доп. ограничения	Необх. подзадачи
1	10	$n \leq 5$	
2	10	$n \leq 8$	1
3	10	$n \leq 10$	1,2
4	10	$n = 11$	1,2,3
5	10	$n = 12$	1,2,3,4
6	11	$n = 13$	1,2,3,4,5
7	12	$n = 14$	1,2,3,4,5
8	13	$n = 15$	1,2,3,4,5
9	14	$n = 16$	1,2,3,4,5

Примеры

стандартный ввод	стандартный вывод
5 1	17 3 3
5 2	23 1 1 3 4

Замечание

В первом примере существует только одно оптимальное расположение ферзя на доске 5×5 .



Во втором примере существует несколько оптимальных расположений, защищающих 23 клетки, например:

