

## Задача А. Голодный еж

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

Еж Матвей имеет сад, разбитый на участки в виде двумерной таблицы; строки и столбцы сада нумеруются с 0.

В саду находятся  $n$  фруктов — либо яблоки, либо груши.  $i$ -й фрукт описывается тройкой  $(x_i, y_i, z_i)$ , где  $x_i$  — номер строки,  $y_i$  — номер столбца, а  $z_i$  — тип фрукта (1 для яблока и 2 для груши). Каждый фрукт находится в центре некоторого участка, и в каждом участке может находиться не более одного фрукта.

Изначально Матвей находится в участке  $(0, 0)$ . В любой момент он может сдвинуться на одну клетку либо вниз по строке, либо вправо по столбцу: из участка  $(x, y)$  он может перейти либо в  $(x + 1, y)$ , либо в  $(x, y + 1)$ .

Матвей, будучи голодным ежом, хочет собрать как можно больше фруктов. Однако, чтобы его рацион оставался сбалансированным, среди всех способов собрать максимальное число фруктов он хочет выбрать такой, где абсолютная разность между количеством собранных яблок и груш минимальна.

### Формат входных данных

В первой строке задано число  $n$  ( $1 \leq n \leq 35000$ ) — количество фруктов в саду Матвея.

В следующих  $n$  строках заданы тройки  $x_i, y_i, z_i$  ( $1 \leq x_i, y_i \leq n, z_i \in \{1, 2\}$ )

### Формат выходных данных

Выведите два целых числа:

- максимальное число фруктов, которое может собрать Матвей;
- минимальную возможную абсолютную разность между количеством собранных яблок и груш среди всех способов собрать максимальное число фруктов.

### Система оценки

Баллы за подзадачу начисляются, только если пройдены все тесты этой подзадачи и всех необходимых для неё подзадач.

Если во всех тестах подгруппы максимальное число фруктов найдено правильно, но минимальная разность найдена неправильно, то за подгруппу дается 30% баллов. Обратите внимание, что даже если минимальная разность найдена неправильно, нужно все равно вывести какое-либо число, иначе такое решение получит «Неправильный ответ»

Подзадача	Баллы	Дополнительные ограничения	Необходимые подзадачи	Информация о проверке
0	0	Тесты из условия	—	полная проверка
1	5	$1 \leq n \leq 100$	—	первая ошибка
2	5	$1 \leq n \leq 400$	1	первая ошибка
3	10	$1 \leq n \leq 3500$	1–2	первая ошибка
4	15	$1 \leq n \leq 5000$	1–3	первая ошибка
5	25	$1 \leq n \leq 15000$	1–4	первая ошибка
6	40	$1 \leq n \leq 35000$	1–5	первая ошибка

## Пример

стандартный ввод	стандартный вывод
4 1 3 1 2 4 1 3 1 2 4 2 2	2 2

## Замечание

Матвей может собирать только те фрукты, до участков с которыми можно добраться, двигаясь из  $(0, 0)$  только вправо и вниз.

## Задача В. Тюршмат

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

Тимофей и Андрей, два лучших игрока в мире, встретятся друг с другом в финале чемпионата мира по игре «Тюршмат».

«Тюршмат» — это игра для двух игроков на последовательности положительных целых чисел. Игроки ходят по очереди, и проигрывает тот, кто не может сделать ход. В самом начале на доске есть только одна последовательность положительных целых чисел — исходная последовательность.

За один ход игрок выбирает **одну** из существующих последовательностей и **одно** число  $x$ , которое встречается в этой последовательности. Затем он удаляет **все вхождения** числа  $x$  из выбранной последовательности, тем самым **разбивая** её на несколько новых последовательностей по позициям, где встречалось число  $x$ .

Была найдена последовательность чисел — шаблон для финального матча. Известно, что исходная последовательность в финале будет некоторым подотрезком этого шаблона. Вам даны  $q$  сценариев. Для каждого сценария определите, кто будет победителем, если и Тимофей, и Андрей играют оптимально, а первым ходит Тимофей.

### Формат входных данных

Первая строка содержит два целых числа  $n$  и  $q$  ( $1 \leq n, q \leq 10^5$ ).

Вторая строка содержит последовательность положительных целых чисел  $a_1, a_2, \dots, a_n$  ( $1 \leq a_i \leq 32$  для каждого  $i = 1, 2, \dots, n$ ).

Каждая из следующих  $q$  строк содержит два целых числа  $l_i$  и  $r_i$  ( $1 \leq l_i \leq r_i \leq n$ ) — границы отрезка, рассматриваемой в  $i$ -м сценарии.

### Формат выходных данных

Для каждого запроса выведите в отдельной строке либо `Timofey`, либо `Andrey` — имя победителя в соответствующем сценарии.

### Система оценки

Подзадача	Баллы	Дополнительные ограничения	Необходимые подзадачи	Информация о проверке
0	0	Тесты из условия	—	полная проверка
1	15	$n, q \leq 10$	—	первая ошибка
2	11	$n, q \leq 1000, a_i \leq 2$	—	первая ошибка
3	18	$n, q \leq 1000$	1–2	первая ошибка
4	14	$a_i \leq 2$	2	первая ошибка
5	13	$a_{l_i} = a_{r_i}$ для всех $i = 1, \dots, q$	—	первая ошибка
6	29	Без дополнительных ограничений	1–5	первая ошибка

## Примеры

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

## Замечание

В первом примере: в третьем сценарии Тимофей выбирает  $x = 2$ , чем разбивает последовательность на две последовательности длины 1. Какую бы последовательность Андрей ни выбрал следующим ходом, Тимофей затем выберет оставшуюся, после чего у Андрея не останется возможных ходов.

## Задача С. Непотопляемая стена

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	4 секунды
Ограничение по памяти:	512 мегабайт

На дворе XIV век, и вскоре начнётся строительство Тракайского островного замка. Первой задачей главного архитектора является проектирование главной крепостной стены.

Построить стену, которая сможет защитить замок от любой возможной атаки, не так просто. Чтобы обеспечить безопасность гарнизона, главный архитектор уже сузил множество допустимых проектов. Поскольку атаки со стороны середины озера менее вероятны, чем со стороны берега, стена не обязана образовывать замкнутый контур. Вместо этого она будет иметь форму прямой линии и состоять из  $n$  башен, расположенных последовательно и пронумерованных от 1 до  $n$ . Остаётся выбрать высоту каждой башни.

Для каждой башни архитектор уже выбрал два возможных значения высоты. Он решил, что высота  $i$ -й башни будет либо  $a_i$ , либо  $b_i$ . Таким образом, всего существует  $2^n$  возможных стен.

Однако у замка, расположенного на небольшом острове посреди озера, есть и свои трудности. Во время сильных штормов замок может затапливать. В таких случаях вода скапливается над башнями стены, если по обе стороны от них находятся более высокие части, не позволяющие воде стечь.

Для конкретного выбора высот башен нас интересует количество воды, которое соберётся на стене после сильной бури.

Формально, для каждого  $i = 1, 2, \dots, n$  уровень воды в позиции  $i$  не ниже  $h$  тогда и только тогда, когда существуют такие целые числа  $l$  и  $r$ , что  $l \leq i \leq r$  и высоты башен в позициях  $l$  и  $r$  не меньше  $h$ .

В частности:

- уровень воды в позициях 1 и  $n$  всегда равен высоте соответствующих частей;
- уровень воды в любой позиции всегда не меньше высоты соответствующей части.

Количество воды, собравшейся в позиции  $i$ , равно разности между уровнем воды и высотой башни в этой позиции. Общее количество собравшейся воды равно сумме по всем позициям от 1 до  $n$ .

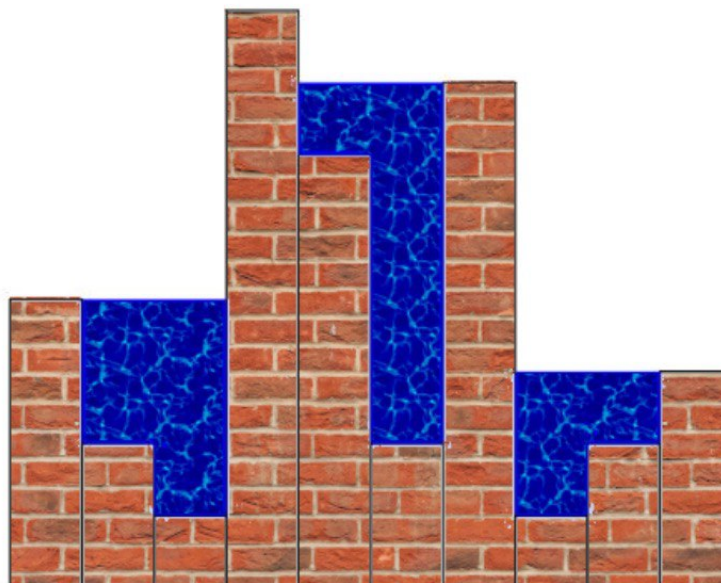


Рис. 1: Иллюстрация стены.

Требуется вычислить сумму общего количества собравшейся воды по всем  $2^n$  возможным стенам. Так как ответ может быть очень большим, выведите его по модулю  $10^9 + 7$ .

### Формат входных данных

Первая строка содержит одно целое число  $n$  ( $1 \leq n \leq 5 \cdot 10^5$ ).

Вторая строка содержит  $n$  целых чисел  $a_1, a_2, \dots, a_n$  ( $1 \leq a_i, b_i \leq 10^9$  для всех  $1 \leq i \leq n$ )

Третья строка содержит  $n$  целых чисел  $b_1, b_2, \dots, b_n$  ( $a_i \neq b_i$  для всех  $1 \leq i \leq n$ )

### Формат выходных данных

Выведите одно целое число — сумму общего количества собравшейся воды по всем  $2^n$  возможным стенам по модулю  $10^9 + 7$ .

### Система оценки

Баллы за подзадачу начисляются, только если пройдены все тесты этой подзадачи и всех необходимых для неё подзадач.

Подзадача	Баллы	Дополнительные ограничения	Необходимые подзадачи	Информация о проверке
0	0	Тесты из условия	—	полная проверка
1	8	$n \leq 20$	—	первая ошибка
2	17	$n \leq 100$ и для всех башен $a_i, b_i \leq 1000$	—	первая ошибка
3	19	$n \leq 10000$ и для всех башен $a_i, b_i \leq 1000$	2	первая ошибка
4	14	$n \leq 10000$	1–3	первая ошибка
5	12	Для всех башен $a_i, b_i \leq 2$	—	первая ошибка
6	30	Без дополнительных ограничений	0–5	первая ошибка

### Примеры

стандартный ввод	стандартный вывод
4 1 1 1 1 2 2 2 2	6
10 1 2 3 4 5 6 7 8 9 10 10 9 8 7 6 5 4 3 2 1	21116

### Замечание

Существует ровно одна возможная стена, для которой собираются две единицы воды:

$$2\ 1\ 1\ 2.$$

Также существуют четыре стены, для которых собирается ровно одна единица воды:

$$1\ 2\ 1\ 2, \quad 2\ 1\ 2\ 1, \quad 2\ 1\ 2\ 2, \quad 2\ 2\ 1\ 2.$$

## Задача D. Капустные магазины

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	5 секунд
Ограничение по памяти:	1024 мегабайта

*Это интерактивная задача.*

Мальчик Петя путешествует по стране и исследует магазины, в которых продают капусту. Всего в стране есть  $n$  магазинов, пронумерованных числами от 0 до  $n - 1$ . Некоторые пары магазинов соединены двусторонними дорогами.

Гарантируется, что:

- Из любого магазина можно добраться в любой другой
- Между одной парой различных магазинов существует не более одной дороги
- От каждого магазина отходит **не более 7 дорог**

Петя хочет восстановить полную схему дорог между магазинами. Для этого он пользуется спутниковой системой наблюдения.

Спутники умеют отвечать на вопрос следующего вида:

Можно ли добраться из одного магазина в другой, если разрешено посещать только некоторые указанные магазины?

Требуется, используя такие запросы, определить все дороги между магазинами.

### Протокол взаимодействия

В начале взаимодействия жюри сообщает вашей программе два целых числа  $t, n$  ( $0 \leq t \leq 5, 2 \leq n \leq 1400$ ) — номер подгруппы и число магазинов.

Чтобы обратиться к спутнику, ваша программа должна вывести строчку:

$$? a b p_0 p_1 \dots p_{n-1}$$

Для запроса должно выполняться:  $0 \leq a < n, 0 \leq b < n$ ,

Каждое  $p_i \in \{0, 1\}$ , **обязательно**  $p_a = p_b = 1$ .

Здесь  $p_i = 1$  означает, что магазин  $i$  разрешено использовать, а  $p_i = 0$  — запрещено.

В ответ программа жюри возвращает одно число:

- 1, если существует путь из магазина  $a$  в магазин  $b$ , проходящий только через магазины  $i$ , для которых  $p_i = 1$ ;
- 0 в противном случае.

Когда ваша программа восстановит все дороги, она должна вывести их в формате

$$! m$$

где  $m$  — число дорог, а затем вывести  $m$  строк вида  $a b$  — по одной на каждую дорогу.

Каждая дорога должна быть выведена ровно один раз, причём должно выполняться  $0 \leq a < b < n$ .

После этого программа должна завершить работу.

Необходимо восстановить все дороги между магазинами, используя не более 45000 запросов к спутниковой системе, вывод ответа не считается за запрос.

Если ваша программа сделает больше запросов, получит неверный формат ответа или выведет неправильный список дорог, она получит вердикт «Неправильный ответ»

Всего дорог между магазинами обозначим через число  $m$ . Гарантируется, что  $1 \leq m \leq 1500$ .

После каждого запроса первого типа необходимо сбрасывать буфер вывода. Например, в C++ это можно сделать с помощью `cout << endl;` или `cout.flush();` Невыполнение сброса буфера может привести к тому, что программа будет ожидать ответ жюри бесконечно долго.

## Система оценки

Баллы за подзадачу начисляются, только если пройдены все тесты этой подзадачи и всех необходимых для неё подзадач.

Подзадача	Баллы	Дополнительные ограничения	Необходимые подзадачи	Информация о проверке
0	0	Тесты из условия	—	полная проверка
1	10	$n \leq 250$	—	первая ошибка
2	10	$m = n - 1$ , дороги у магазинов представляет собой путь	—	первая ошибка
3	27	$m = n - 1$ , для каждого $i$ ( $1 \leq i \leq n - 1$ ) из магазина 0 можно добраться до магазина $i$ , пройдя не более чем через 8 других магазинов	—	первая ошибка
4	30	$m = n - 1$	2–3	первая ошибка
5	23	Без дополнительных ограничения	0–4	первая ошибка

## Пример

стандартный ввод	стандартный вывод
1 6	? 0 1 1 1 0 0 0 0
1	? 0 2 1 0 1 0 0 0
0	? 0 5 1 0 1 0 0 1
0	? 0 5 1 1 1 0 0 1
1	? 2 3 1 1 1 1 1 1
1	! 7
	1 0
	4 1
	2 4
	3 4
	2 5
	1 2
	0 3

## Замечание

Описанное в примере существует лишь для показания формата взаимодействия с интерактором, оно не обязательно отражает правильное решение для данного примера.

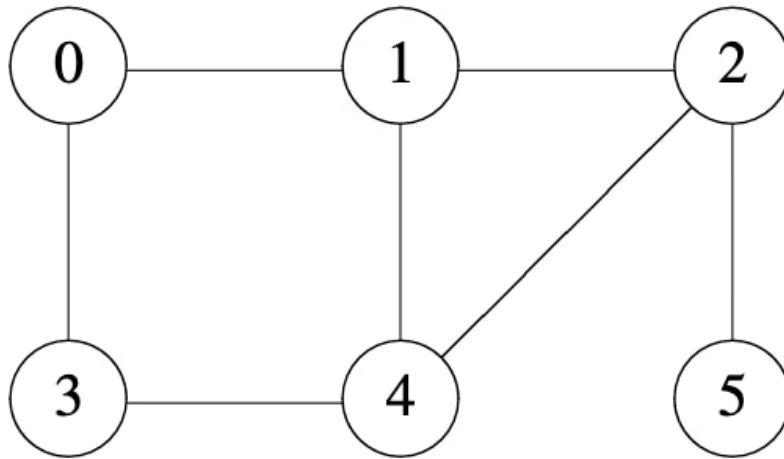


Рис. 1: Граф из примера.