

## Задача А. К-ый максимум

Имя входного файла: `kthmax.in`  
Имя выходного файла: `kthmax.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

Напишите программу, реализующую структуру данных, позволяющую добавлять и удалять элементы, а также находить  $k$ -й максимум.

### Формат входных данных

Первая строка входного файла содержит натуральное число  $n$  — количество команд ( $n \leq 100\,000$ ). Последующие  $n$  строк содержат по одной команде каждая. Команда записывается в виде двух чисел  $c_i$  и  $k_i$  — тип и аргумент команды соответственно ( $|k_i| \leq 10^9$ ). Поддерживаемые команды:

- $+1$  (или просто  $1$ ): Добавить элемент с ключом  $k_i$ .
- $0$ : Найти и вывести  $k_i$ -й максимум.
- $-1$ : Удалить элемент с ключом  $k_i$ .

Гарантируется, что в процессе работы в структуре не требуется хранить элементы с равными ключами или удалять несуществующие элементы. Также гарантируется, что при запросе  $k_i$ -го максимума, он существует.

### Формат выходных данных

Для каждой команды нулевого типа в выходной файл должна быть выведена строка, содержащая единственное число —  $k_i$ -й максимум.

### Примеры

<code>kthmax.in</code>	<code>kthmax.out</code>
11	7
+1 5	5
+1 3	3
+1 7	10
0 1	7
0 2	3
0 3	
-1 5	
+1 10	
0 1	
0 2	
0 3	

## Задача В. Своппер

Имя входного файла: `swapper.in`  
Имя выходного файла: `swapper.out`  
Ограничение по времени: 4 секунды  
Ограничение по памяти: 256 мегабайт

Современные компьютеры зацикливаются  
в десятки раз эффективнее человека

Рекламный проспект OS Vista-N

Перед возвращением в штаб-квартиру корпорации Аазу и Скиву пришлось заполнить на местной таможне декларацию о доходах за время визита. Получилась довольно внушительная последовательность чисел. Обработка этой последовательности заняла весьма долгое время.

— Своппер кривой, — со знанием дела сказал таможенник.

— А что такое своппер? — спросил любопытный Скив.

Ааз объяснил, что своппер — это структура данных, которая умеет делать следующее.

- Взять отрезок чётной длины от  $x$  до  $y$  и поменять местами число  $x$  с  $x + 1$ ,  $x + 2$  с  $x + 3$ , и т.д.
- Посчитать сумму чисел на произвольном отрезке от  $a$  до  $b$ .

Учитывая, что обсчёт может затянуться надолго, корпорация «МИФ» попросила Вас решить проблему со своппером и промоделировать ЭТО эффективно.

### Формат входных данных

Во входном файле заданы один или несколько тестов. В первой строке каждого теста записаны число  $N$  — длина последовательности и число  $M$  — число операций ( $1 \leq N, M \leq 100\,000$ ). Во второй строке теста содержится  $N$  целых чисел, не превосходящих  $10^6$  по модулю — сама последовательность. Далее следуют  $M$  строк — запросы в формате 1  $x_i$   $y_i$  — запрос первого типа, и 2  $a_i$   $b_i$  — запрос второго типа. Сумма всех  $N$  и  $M$  по всему файлу не превосходит 200 000. Файл завершается строкой из двух нулей. Гарантируется, что  $x_i < y_i$ , а  $a_i \leq b_i$ .

### Формат выходных данных

Для каждого теста выведите ответы на запросы второго типа, как показано в примере. Разделяйте ответы на тесты пустой строкой.

### Примеры

<code>swapper.in</code>	<code>swapper.out</code>
5 5	Swapper 1:
1 2 3 4 5	10
1 2 5	9
2 2 4	2
1 1 4	
2 1 3	
2 4 4	
0 0	

## Задача С. Переворот

Имя входного файла: `reverse.in`  
Имя выходного файла: `reverse.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

Дан массив. Надо научиться обрабатывать два типа запросов.

- 1 L R - перевернуть отрезок [L, R]
- 2 L R - найти минимум на отрезке [L, R]

### Формат входных данных

Первая строка файла содержит два числа  $n, m$ . ( $1 \leq n, m \leq 10^5$ ) Во второй строке находится  $n$  чисел  $a_i$  ( $1 \leq a_i \leq 10^9$ )- исходный массив. Остальные  $m$  строк содержат запросы, в формате описанном в условии. Для чисел L,R выполняется ограничение ( $1 \leq L \leq R \leq n$ ).

### Формат выходных данных

На каждый запрос типа 2, во входной файл выведите ответ на него, в отдельной строке.

### Примеры

<code>reverse.in</code>	<code>reverse.out</code>
10 7	3
5 3 2 3 12 6 7 5 10 12	2
2 4 9	2
1 4 6	2
2 1 8	
1 1 8	
1 8 9	
2 1 7	
2 3 6	

## Задача D. Легенды и мифы Южного Бутово

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

В 3141 году Бутово превратилось в очень опасный район, полный убийц и прочих жуликов. Настолько опасный, что там стало страшно перемещаться даже на танке!

Напомним, что Бутово состоит из  $N$  проспектов, идущих по направлению с севера на юг, и  $M$  улиц, идущих с востока на запад. Каждый проспект пересекается с каждой улицей ровно на одном перекрестке. Перемещение вдоль проспекта или улице на высокой скорости безопасно, а поворот где-либо, наоборот, очень опасен, так как для этого приходится снизить скорость, и в тот же миг вас начинают атаковать группировки местных жителей.

Управление полицией Бутово решило немного исправить ситуацию. Руководители управления решили, что можно поставить несколько постов на некоторых перекрестках, чтобы на них жители могли спокойно поворачивать, не опасаясь быть атакованными, так как полиция уже взяла на вооружение новейшие винтовки «Мушкет-1812» и, в случае чего, сможет защитить честь и достоинство каждого законопослушного гражданина.

К сожалению, именно в этом году начальник управления решил построить себе новый элитный особняк, поэтому управление решило потратить как можно меньше денег на строительство новых постов. Однако, чтобы все выглядело чистенько, необходимо построить посты так, чтобы можно было добраться от любого перекрестка до любого другого, поворачивая лишь на специально оборудованных постах. Иначе нагрянет проверка и всех уволят.

На этом проблемы Бутово не закончились. Некоторые районы Бутово являются более опасными, и посты, которые будут построены в более опасных районах, стоят дороже. У каждого района есть центр, который находится возле некоторого перекрестка. Любой другой перекресток принадлежит району, центр которого является ближайшим к перекрестку. Расстояние между перекрестками измеряется с помощью так называемого Бутовского расстояния: расстояние между двумя перекрестками — это минимальное количество промежуточных перекрестков, которое нужно преодолеть, чтобы попасть из одного в другой, плюс один. Если же существует несколько районных центров, находящихся на минимальном расстоянии от некоторого перекрестка, то на этом перекрестке постоянно происходят стычки двух авторитетов, поэтому на этом перекрестке построить пост невозможно.

Вам требуется, зная расположение районов Бутово, определить минимальную стоимость постройки постов так, чтобы можно было безопасно добраться от любого перекрестка до любого другого.

### Формат входных данных

Первая строка входного файла содержит три числа  $N$ ,  $M$ ,  $R$  — количество проспектов, количество улиц и количество районов в Бутово, соответственно ( $2 \leq N, M \leq 500$ ,  $1 \leq R \leq 1000$ ). Следующие  $R$  строк содержат по три целых числа — номер проспекта и номер улицы, на которых расположен центр соответствующего района, и стоимость постройки поста в этом районе. Проспекты и улицы нумеруются с единицы, стоимость постройки всюду положительна и не превосходит одной тысячи. Никакие два района не имеют совпадающих центров.

### Формат выходных данных

На первой строке выходного файла выведите два числа: количество перекрестков  $K$ , на которых стоит разместить посты, и найденную минимальную суммарную стоимость постройки. На следующих  $K$  строках должны содержаться описания найденных перекрестков: номер проспекта и номер улицы, на пересечении которых расположен соответствующий перекресток. Если посты нужным образом разместить невозможно, выведите в выходной файл единственное число  $-1$

## Примеры

police.in	police.out
4 3 2 3 1 200 1 3 150	6 1050 2 3 1 3 1 2 4 1 4 2 3 2
3 3 2 1 2 200 3 2 150	-1