

Задача А. Сортировка через один

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 5 секунд
Ограничение по памяти: 256 мегабайт

Напишите программу, которая сортирует по возрастанию все элементы массива с нечётными номерами, а все элементы с чётными номерами – по убыванию. Нумерация элементов массива начинается с единицы.

Формат входных данных

Первая строка содержит размер массива N . Во второй строке через пробел задаются N чисел – элементы массива. Гарантируется, что $0 < N \leq 10000$. Все элементы массива не превосходят по модулю 10^6 .

Формат выходных данных

Программа должна вывести все элементы отсортированного массива в одну строку, разделив их пробелами.

Пример

стандартный ввод	стандартный вывод
6	2 5 4 3 6 1
6 1 4 3 2 5	

Задача В. Сортировка половин

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 5 секунд
Ограничение по памяти: 64 мегабайта

Напишите программу, которая сортирует первую половину массива по возрастанию, а вторую – по убыванию. При этом элементы из первой половины не должны перемещаться во вторую и наоборот.

Формат входных данных

Первая строка содержит размер массива N . Во второй строке через пробел задаются N чисел – элементы массива. Гарантируется, что $0 < N \leq 10000$ и N – чётное число. Все числа помещаются в Integer.

Формат выходных данных

Программа должна вывести в одной строке элементы все массива, отсортированного нужным образом, разделив их пробелами.

Пример

стандартный ввод	стандартный вывод
6 3 2 1 4 5 6	1 2 3 6 5 4

Задача С. Упаковка Груза

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

Не за горами очередная олимпиада, на которой Сережа сможет получить очередную футболку, поэтому самое время собирать вещи! Среди прочих вещей Сережа собирается взять с собой невероятно ценный груз — игровой ноутбук, весящий 10 килограммов. Очевидно, его нужно упаковать таким образом, чтобы он не сломался в багажном отделении самолета.

Для упаковки ноутбука Сережа купил N коробок разного размера. Размер i -й коробки равен s_i . Известно, что ноутбук можно вложить в каждую коробку без особых проблем, однако Сережа очень переживает за столь ценный груз, поэтому решил упаковать ноутбук более надежным способом: упаковать ноутбук в некоторую коробку, которую упаковать в некоторую другую коробку, и так далее...

При упаковке одной коробки в другую важно заполнить пустое место уплотнителем, ведь иначе ноутбук, лежащий где-то внутри, может сломаться! Формально, i -ю коробку можно упаковать внутрь j -й коробки, если $\frac{s_j}{s_i} \geq K$. Как уже было написано выше, Сережа очень дорожит ноутбуком, поэтому хочет упаковать его в наибольшее количество вложенных друг в друга коробок.

От вас требуется найти такое максимальное L , что можно выбрать L коробок из имеющихся, таким образом, чтобы каждую следующую коробку можно было вложить в предыдущую.

Формат входных данных

В первой строке записаны целые числа N и K ($1 \leq N \leq 10^5$, $2 \leq K \leq 10^9$) — количество коробок и минимальное отношение размеров коробок, которые можно вложить друг в друга.

Во второй строке записаны N целых чисел s_1, s_2, \dots, s_N ($1 \leq s_i \leq 10^9$) — размеры коробок.

Формат выходных данных

Выведите одно число — максимальное число L , такое что можно выбрать L коробок таким образом, что каждую следующую коробку можно вложить в предыдущую.

Система оценки

Подзадача	Баллы	Дополнительные ограничения	Оценка	Необходимые подзадачи
0	0	Тесты из условия	подзадача	—
1	20	$1 \leq N \leq 20$	подзадача	—
2	20	$K = 10^9$	подзадача	—
3	60	—	подзадача	1, 2

Примеры

стандартный ввод	стандартный вывод
5 2 1 8 4 2 100	5
3 2 2 3 2	1

Замечание

В первом примере можно взять все коробки и упаковать их следующим образом: $1 \rightarrow 2 \rightarrow 4 \rightarrow 8 \rightarrow 100$.

Во втором примере можно взять не более одной любой коробки.

Задача D. Олимпиада

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

Это задача на программирование. Решением является код, написанный на одном из предложенных языков программирования.

Представьте, что вы находитесь на олимпиаде по информатике (знакомое ощущение, правда?), до конца тура осталось всего T минут, а у вас еще куча нерешенных задач... Целых N штук!

Для каждой задачи известно, сколько баллов вы получите в случае, если решите ее. За решение первой задачи вы получите a_1 баллов, за решение второй задачи — a_2 баллов, и так далее. Так как вы очень долго и упорно готовились к олимпиаде, вы точно знаете, что способны решить любую задачу за C минут. Поэтому единственное, что остается сделать, это выбрать, какие задачи нужно решить, чтобы получить как можно больше баллов.

Разумеется, задачи можно решать только на протяжении оставшихся T минут. Также будем считать, что если вы начали решать некоторую задачу, вы не можете прерваться до тех пор, пока не решите ее.

Формат входных данных

В первой строке записаны три целых числа: T и C ($1 \leq T, C \leq 10^9$) — время, оставшееся до конца тура, а также время, необходимое для решения любой задачи.

Во второй строке записано целое число N ($1 \leq N \leq 10^5$) — количество задач.

В третьей строке через пробел записаны N целых чисел: a_1, a_2, \dots, a_N ($1 \leq a_i \leq 1\,000$), которые означают, что вы получите a_i баллов, если решите i -ю задачу.

Формат выходных данных

Выведите одно число — максимальное количество баллов, которые вы сможете набрать за оставшееся время.

Система оценки

Подзадача	Баллы	Дополнительные ограничения	Оценка	Необходимые подзадачи
0	0	Тесты из условия	подзадача	—
1	30	$1 \leq N \leq 20$	подзадача	—
2	30	$1 \leq N \leq 1\,000$	подзадача	1
3	40	—	подзадача	1, 2

Примеры

стандартный ввод	стандартный вывод
180 42 6 100 100 100 100 100 100	400
10 9 5 10 50 2 100 36	100
100 42 6 4 8 15 16 23 42	65
10 20 1 100	0

Замечание

В первом примере можно успеть решить любые четыре задачи и получить 400 баллов.

Во втором примере можно успеть решить лишь одну задачу.

В четвертом примере нельзя успеть решить ни одну задачу.

Задача E. Дипломы в папках

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

В этом году Иван заканчивает школу и поступает в вуз. За время своей учебы он часто участвовал в олимпиадах по информатике и у него накопилось много дипломов. Иван раскладывал дипломы по папкам совершенно бессистемно, то есть любой диплом мог оказаться в любой из папок. К счастью, Иван помнит, сколько дипломов лежит в каждой из папок.

Иван хочет принести в приемную комиссию выбранного вуза папку, в которой находится диплом Московской олимпиады по программированию (такой диплом у Ивана ровно один). Для того чтобы понять, что в данной папке нужного диплома нет, Ивану нужно просмотреть все дипломы из этой папки. Просмотр одного диплома занимает у него ровно одну секунду и он может мгновенно переходить к просмотру следующей папки после окончания просмотра предыдущей. Порядок просмотра папок Иван может выбирать.

По заданному количеству дипломов в каждой из папок требуется определить, за какое наименьшее время в худшем случае Иван поймет, в какой папке содержится нужный ему диплом.

Формат входных данных

В первой строке входного файла записано целое число N ($1 \leq N \leq 100$) — количество папок. Во второй строке записаны N целых чисел a_1, a_2, \dots, a_N ($1 \leq a_i \leq 100$) — количество дипломов в каждой из папок.

Формат выходных данных

Выведите одно число — минимальное количество секунд, необходимое Ивану в худшем случае для определения того, в какой папке содержится диплом.

Пример

стандартный ввод	стандартный вывод
2 2 1	1

Задача F. Лука и локальная сеть динозавров

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

Лука смог приобрести всю коллекцию динозавров из «Шестерочки» и обнаружил, что в динозаврах есть коммутаторы, поэтому ему захотелось их объединить в одну локальную сеть.

Он расставил на декартовой плоскости всю коллекцию, то есть местоположение каждого динозавра задано двумя числами — координатами по оси x и y .

Лука хочет соединить их проводами так, чтобы между любыми двумя динозаврами существовал путь по этим проводам. Луку раздражают спутанные провода, поэтому никакие два провода не должны пересекаться (пересечения в концах отрезков разрешены). Кроме того, у Луки мало денег на покупку проводов, поэтому общее количество проведенных отрезков не должно превышать $2n$.

Формат входных данных

Первая строка содержит одно целое число n ($1 \leq n \leq 10^3$) — количество динозавров.

В следующих $2n$ строках заданы координаты динозавров. В каждой строке записано одно целое число: первая строка содержит координату по оси x для первого динозавра, вторая строка — координату по оси y для первого динозавра, третья строка — координату по оси x для второго динозавра, и так далее. Таким образом координата x_i для i -го динозавра находится $(2i)$ -й строке входных данных, координата y_i для i -го динозавра находится в $(2i + 1)$ -й строке входных данных. Гарантируется, что $(-10^9 \leq x_i, y_i \leq 10^9)$, а также никакие два динозавра не находятся в одной точке плоскости.

Формат выходных данных

В первой строке выведите одно число m — количество проведенных проводов, либо число -1 , если соединить динозавров описанным в условии способом невозможно.

Если существует подходящее под условие соединение, то в следующих m строках выведите по два целых числа — порядковые номера динозавров, соединенных очередным проводом.

Если решений несколько, можно вывести любое из них.

Система оценки

Решения, правильно работающие только для случаев, когда n не превосходит 4, будут оцениваться в 25 баллов.

Решения, правильно работающие только для случаев, когда никакие три динозавра не лежат на одной прямой, будут оцениваться в 25 баллов.

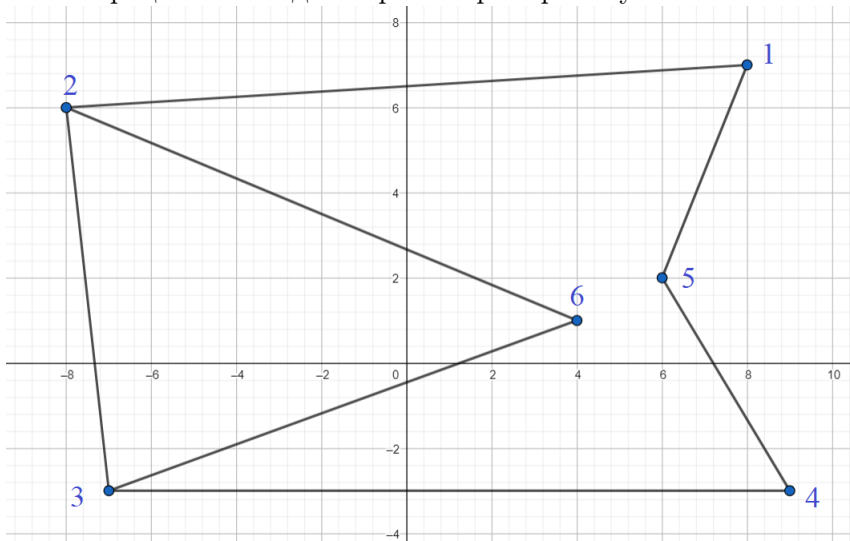
Решения, правильно работающие только для случаев, когда у всех динозавров координаты по оси x различны, будут оцениваться в 25 баллов.

Примеры

стандартный ввод	стандартный вывод
6 8 7 -8 6 -7 -3 9 -3 6 2 4 1	7 1 2 1 5 2 3 2 6 3 6 3 4 4 5
3 -3 4 8 -4 -1 0	2 1 3 3 2

Замечание

Иллюстрация ответа для первого примера из условия:



Задача G. Университетская команда

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

Ваш университет всюю готовится к новому сезону командных олимпиад. Как несложно догадаться, для этого необходимо собрать команду, которая будет представлять учебное заведение.

Всего в университете учатся n студентов, i -й из них имеет силу a_i . Для участия в олимпиадах необходимо выбрать **ровно** k людей и расположить их в каком-то порядке. Пусть были выбраны студенты с номерами i_1, i_2, \dots, i_k (именно в таком порядке). Тогда *слабость* команды равна $|a_{i_2} - a_{i_1}| + |a_{i_3} - a_{i_2}| + \dots + |a_{i_k} - a_{i_{k-1}}|$. Иначе говоря, слабость команды равна сумме модулей разностей сил соседних участников команды, если их расположить в выбранном порядке. **Обратите внимание, что порядок студентов вы выбираете сами.**

Администрация университета просит вас определить минимально возможную слабость команды.

Формат входных данных

Первая строка содержит одно целое число n ($1 \leq n \leq 100\,000$) — количество студентов, обучающихся в университете.

Вторая строка содержит одно целое число k ($1 \leq k \leq n$) — количество людей в составе команды.

Каждая из следующих n строк содержит целое число a_i ($1 \leq a_i \leq 10^9$) — силу i -го студента.

Формат выходных данных

Выведите одно целое число — минимально возможную слабость команды.

Система оценки

Решения, правильно работающие при $n \leq 10$, будут оцениваться в 25 баллов.

Решения, правильно работающие при $n \leq 1\,000$, будут оцениваться в 50 баллов.

Решения, правильно работающие при $a_i \leq 1\,000\,000$, будут оцениваться в 50 баллов.

Пример

стандартный ввод	стандартный вывод
5 3 1 5 3 2 1	1

Замечание

Рассмотрим пример. Если взять студентов с номерами 1, 5 и 4 (именно в таком порядке), то слабость команды будет равна $|1 - 1| + |2 - 1| = 0 + 1 = 1$. Можно доказать, что меньшую слабость получить не получится.

Задача Н. Удаление данных

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 512 мегабайт

Случилась беда — шпиона Сергея раскрыли, и теперь ему нужно срочно бежать! Но перед побегом он должен удалить все компрометирующие данные со своего компьютера.

На компьютере Сергея сохранены N файлов, пронумерованных числами от 1 до N . У каждого из файлов есть размер в байтах: a_1, a_2, \dots, a_N . Все данные на компьютере Сергея хорошо зашифрованы. Шпион определил, что для удаления файла с номером i понадобится минимум из a_{i-1} и a_{i+1} секунд (для удаления первого файла потребуется a_2 секунд, а для удаления последнего — a_{N-1} секунд). Когда остается всего один файл, он удаляется мгновенно. После удаления файла с номером i остальные файлы перенумеровываются последовательно.

У Сергея осталось очень мало времени, а ему еще нужно собрать вещи, поэтому он просит у вас помощи. Определите, какое минимальное время понадобится шпиону, чтобы удалить все файлы. Сергей может удалять файлы последовательно в любом порядке.

Формат входных данных

В первой строке выходных данных записано одно целое число N ($1 \leq N \leq 10^5$) — количество файлов на компьютере шпиона.

В каждой из следующих N строк записано по одному целому числу a_i ($1 \leq a_i \leq 10^4$) — размер файла с номером i на компьютере Сергея.

Формат выходных данных

В единственной строке выведите одно число — минимальное время, которое понадобится Сергею для удаления всех файлов.

Система оценки

Решения, правильно работающие только для случаев, когда N не превосходит 10, будут оцениваться в 20 баллов.

Решения, правильно работающие только для случаев, когда N не превосходит 1 000, будут оцениваться в 60 баллов.

Примеры

стандартный ввод	стандартный вывод
5 1 2 3 1 100	4
1 1	0

Замечание

В первом примере у Сергея есть файлы с размерами 1, 2, 3, 1, 100. Один из вариантов решения приведен ниже:

1. Удалим последний файл. Это займет одну секунду.
2. Затем удалим файл размера 2 за одну секунду.
3. Далее удалим файл размера 3 за одну секунду.
4. Теперь удалим любой из оставшихся двух файлов за одну секунду.

5. Последний файл моментально удалится сам.

Итого, Сергею понадобится $1 + 1 + 1 + 1 = 4$ секунды.

Во втором примере у Сергея изначально есть всего один файл, который сразу же удалится.

Задача I. Юные следопыты

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

Отряд юных следопытов отправился в учебную экспедицию навстречу своим первым приключениям. И возглавляет их старший следопыт Рассел. Вот герои зашли в лес, разбили лагерь и дальше решили разделить на группы, чтобы исследовать как можно больше интересных мест. Рассел должен был выбрать состав групп, но столкнулся с одной проблемой. . .

Многие юные следопыты неопытны, и отправлять их маленькими группами — не всегда хорошая идея. Даже сам Рассел недавно стал старшим следопытом и нечасто бывал в экспедициях. Каждый следопыт характеризуется своей неопытностью — целым положительным числом e_i . Рассел решил, что юный следопыт с неопытностью e может идти лишь в группе, количество следопытов в которой не меньше e .

Теперь задача Рассела — определить, какое наибольшее число групп следопытов он сможет организовать. При этом может получиться, что некоторые следопыты не войдут в состав ни одной группы, это не страшно, ведь и в лагере для них найдется работа. Рассел очень переживает за успех экспедиции, и потому попросил вас помочь ему.

Формат входных данных

В первой строке записано число T ($1 \leq T \leq 2 \cdot 10^5$) — количество независимых тестовых случаев. В следующих $2T$ строках следует описание тестовых случаев.

В первой строке описания каждого теста задано целое число юных следопытов N ($1 \leq N \leq 2 \cdot 10^5$).

В следующей строке записаны N целых чисел e_1, e_2, \dots, e_N ($1 \leq e_i \leq N$), где e_i — неопытность i -го следопыта.

Гарантируется, что сумма N по всем тестовым случаям не превосходит $3 \cdot 10^5$.

Формат выходных данных

Выведите T чисел, каждое на отдельной строке.

В i -й строке выведите наибольшее число групп, которое можно организовать в i -м тестовом случае.

Пример

стандартный ввод	стандартный вывод
2	3
3	2
1 1 1	
5	
2 3 1 2 2	

Замечание

В первом примере можно сформировать три группы, в каждой из которых будет один следопыт. Это возможно, так как неопытность всех трех следопытов равна 1, что не меньше, чем размер их групп.

Во втором примере можно сформировать две группы. В первой группе окажутся следопыты с неопытностью 1, 2 и 3, а во второй группе — два следопыта с неопытностью 2.

Этот способ — не единственный возможный. Можно, например, сформировать одну группу из трех следопытов с неопытностью 2, а также еще одну группу, в которой будет всего один следопыт с неопытностью 1. При таком разбиении на группы следопыт с неопытностью 3 не войдет в состав ни одной группы.

Задача J. Фокусы

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

Солдат прибыл в чудесный город и остановился на самом дорогом постоялом дворе. Слуга, который чистил приезжим обувь рассказал обо всех чудесах города, а также о короле и о прекрасной принцессе. Солдату очень захотелось её увидеть, но слуга сказал, что это совсем не просто.

К счастью, солдат знает n фокусов, которые он может показывать в любом порядке. Для того, чтобы продемонстрировать i -й фокус, солдат должен иметь репутацию хотя бы a_i . После демонстрации i -го фокуса репутация солдата изменяется на b_i . Изначально репутация солдата равна нулю.

Солдат — порядочный фокусник, поэтому у всех его фокусов есть очень интересное свойство: если $b_i < b_j$, то $a_i \leq a_j$ для любых i, j .

Солдату понравилась прелестная принцесса, и он хочет показать ей как можно больше фокусов. Помогите ему и определите максимальное количество фокусов, которые он сможет показать.

Формат входных данных

В первой строке вводит натуральное число n ($1 \leq n \leq 2 \cdot 10^5$).

В следующих n строках записаны по два целых числа: a_i и b_i ($|a_i|, |b_i| \leq 10^9$).

Гарантируется, что если $b_i < b_j$, то $a_i \leq a_j$.

Формат выходных данных

Вывести единственное число — максимальное количество фокусов, которые может показать солдат.

Примеры

стандартный ввод	стандартный вывод
2 1 1 1 2	0
4 2 3 -1 -1 -2 -1 5 7	2
4 0 2 -1 1 4 4 3 2	4

Замечание

В первом примере солдат не может показать ни одного фокуса, так как каждый фокус требует репутации хотя бы 1, но изначально репутация солдата равна нулю.

Один из оптимальных вариантов для солдата в третьем примере следующий: первым делом солдат покажет второй фокус, тогда его репутация станет равна 1. Далее солдат может показать первый фокус, после чего его репутация изменится на 2 и станет равна 3. Теперь солдат может показать четвёртый фокус, заработав при этом два очка репутации. Теперь его репутация равна 5, и солдат может показать третий фокус, после чего его репутация увеличится на 4 и станет равна 9.