

Задача А. Скобки

Имя входного файла:	brackets.in
Имя выходного файла:	brackets.out
Ограничение по времени:	1 секунда
Ограничение по памяти:	64 мегабайта

Требуется определить, является ли правильной данная последовательность круглых, квадратных и фигурных скобок.

Формат входного файла

В единственной строке входного файла записано подряд N скобок ($1 \leq N \leq 255$).

Формат выходного файла

В выходной файл вывести «YES», если данная последовательность является правильной, и «NO» в противном случае.

Пример

brackets.in	brackets.out
([])	YES
(({}))	NO

Задача В. Постфиксная запись

Имя входного файла:	postfix.in
Имя выходного файла:	postfix.out
Ограничение по времени:	1 секунда
Ограничение по памяти:	64 мегабайта

В постфиксной записи (или обратной польской записи) операция записывается после двух операндов. Например, сумма двух чисел A и B записывается как $A B +$. Запись $B C + D *$ обозначает привычное нам $(B + C) * D$, а запись $A B C + D * +$ означает $A + (B + C) * D$. Достоинство постфиксной записи в том, что она не требует скобок и дополнительных соглашений о приоритете операторов для своего чтения.

Дано выражение в обратной польской записи. Определите его значение.

Формат входного файла

В единственной строке записано выражение в постфиксной записи, содержащее однозначные числа и операции $+$, $-$, $*$. Строка содержит не более 100 чисел и операций.

Формат выходного файла

Необходимо вывести значение записанного выражения. Гарантируется, что результат выражения, а также результаты всех промежуточных вычислений по модулю меньше 2^{31} .

Пример

postfix.in	postfix.out
8 9 + 1 7 - *	-102

Задача С. Очередь

Имя входного файла:	queue.in
Имя выходного файла:	queue.out
Ограничение по времени:	1 секунда
Ограничение по памяти:	64 мегабайта

Реализуйте работу очереди. Для каждой операции изъятия элемента выведите ее результат.

На вход программе подаются строки, содержащие команды. Каждая строка содержит одну команду. Команда — это либо $+$ N , либо $-$. Команда $+$ N означает добавление в очередь числа N , по модулю не превышающего 10^9 . Команда $-$ означает изъятие элемента из очереди.

Формат входного файла

В первой строке содержится количество команд — M ($1 \leq M \leq 10^6$). В последующих строках содержатся команды, по одной в каждой строке.

Формат выходного файла

Выведите числа, которые удаляются из очереди, по одному в каждой строке. Гарантируется, что изъятий из пустой очереди не производится.

Пример

queue.in	queue.out
4	1
+ 1	10
+ 10	
-	
-	

Задача D. Результаты олимпиады

Имя входного файла:	olymp.in
Имя выходного файла:	olymp.out
Ограничение по времени:	1 секунда
Ограничение по памяти:	64 мегабайта

N участников олимпиады получили уникальные номера от 1 до N . В результате решения задач на олимпиаде каждый участник получил некоторое количество баллов (целое число от 0 до 600). Известно, кто сколько баллов набрал.

Требуется перечислить участников олимпиады в порядке невозрастания набранных ими баллов.

Формат входного файла

В первой строке входного файла содержится число N ($0 \leq N \leq 101$). Далее записаны через пробел N чисел — количество набранных участниками баллов (1-е число — это баллы, набранные участником номер 1, 2 — участником номер 2 и т.д.).

Формат выходного файла

В выходной файл следует вывести N чисел — номера участников в порядке невозрастания набранных ими баллов (участники, набравшие одинаковое количество баллов могут быть выведены в любом порядке).

Пример

olymp.in	olymp.out
5	5 2 4 1 3
100 312 0 312 500	

Задача E. Станция «Сортировочная»

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

На железнодорожной станции «Сортировочная» на пу-

ти находятся n товарных вагонов, из которых необходимо сформировать состав. Все вагоны имеют одинаковую длину, однако в них размещены различные грузы, поэтому вагоны могут иметь разную массу. Работники железнодорожной станции «Сортировочная» должны выстроить вагоны в порядке возрастания масс — тогда составу будет разрешено отправиться в путь.

Обычно для таких целей используются так называемые маневровые тепловозы и электровозы, однако на этой станции ведутся испытания экспериментального устройства для сортировки вагонов. Предполагается, что оно позволит существенно сократить затраты времени на формирование составов.

Это устройство на воздушной подушке перемещается над вагонами, его длина немного превышает длину двух вагонов. Оно может зависнуть над двумя соседними вагонами, поднять их оба в воздух и поменять местами. Однако, грузоподъемность устройства ограничена — указанную операцию оно может выполнить только, если суммарная масса двух вагонов не превышает M .

Ваша задача состоит в том, чтобы написать программу, которая определит, возможно ли с помощью экспериментального устройства для сортировки вагонов расставить вагоны, находящиеся на пути, в необходимом порядке.

Формат входного файла

Первая строка входного файла содержит два числа: число вагонов n ($2 \leq n \leq 100\,000$) и грузоподъемность экспериментального устройства M ($1 \leq M \leq 10^9$). Вторая строка входного файла содержит массы вагонов m_1, \dots, m_n (для этих масс выполняются неравенства $1 \leq m_i \leq 10^9$, кроме этого массы вагонов попарно различны). Массы вагонов перечислены во входном файле в том порядке, в котором вагоны исходно стоят на пути.

Формат выходного файла

В выходной файл выведите слово «Yes», если с помощью экспериментального устройства для сортировки вагонов можно расставить вагоны в необходимом порядке, и слово «No» — иначе.

Примеры

sorting.in	sorting.out
4 10 5 6 3 4	Yes
4 9 5 6 3 4	No

Задача F. Золотой песок

Имя входного файла: dust.in
Имя выходного файла: dust.out
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Во время ограбления магазина вор обнаружил N ящичков с золотым песком. В ящичек под номером i песок имеет стоимость v_i и вес w_i . Чтобы унести награбленное, вор использует рюкзак. Требуется определить наибольшую суммарную стоимость песка, который может унести грабитель, если грузоподъемность рюкзака ограничена величиной W .

Из ящичков можно пересыпать любое количество песка, тогда отношение стоимости отсыпанного песка к стоимости всего ящичка будет равна отношению объема пересыпанного песка к объему всего ящичка.

Формат входного файла

В первой строке входного файла записаны два числа — N и W ($1 \leq N \leq 1000$, $1 \leq W \leq 10^6$). Далее следует N строк по два целых числа в каждой. В i -ой строке записана стоимость v_i и вес w_i песка в i -ом ящичке. Все числа неотрицательные и не превосходят 10^6 .

Формат выходного файла

Выведите искомую максимальную стоимость с точностью до 3 знаков после запятой.

Пример

dust.in	dust.out
3 50 60 20 100 50 120 30	180.000

Задача G. ЛКОИ

Имя входного файла: lkoi.in
Имя выходного файла: lkoi.out
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайт

Лкшатская Командная Олимпиада по информатике (ЛКОИ) проходила согласно таким необычным правилам. Было N команд и T задач. Каждая задача оценивалась с использованием всего лишь одного теста. Таким образом, каждую задачу можно было либо решить полностью, либо не решить совсем.

Количество баллов, назначенное каждой задаче, определялось после соревнования, и было равно количеству команд, которые не решили задачу. Баллы каждой команды подсчитывались как сумма баллов, назначенных задачам, решенным этой командой.

Бурундуки участвовали в соревновании, но они запутались в сложных правилах оценивания, и сейчас они, глядя на результаты, не в состоянии определить свое место в финальном протоколе. Помогите Бурундукам написать программу, которая подсчитает их баллы и место в финальном протоколе.

Перед соревнованием командам присвоили уникальные номера от 1 до N включительно. Номер Бурундуков обозначим P . В финальном протоколе команды перечислены в порядке невозрастания набранных ими баллов. В случае равенства баллов, первыми будут перечислены команды, которые решили больше задач. В случае равенства количества решенных задач, команды с одинаковыми результатами будут перечислены в порядке возрастания их номеров.

Напишите программу, которая по заданной информации о том, какие задачи были решены какими командами, определит количество баллов у Бурундуков и их место в финальном протоколе.

Формат входного файла

Первая строка содержит целые числа N , T и P ($1 \leq N, T \leq 1000$, $1 \leq P \leq N$), разделенные пробелами. После-

дующие N строк описывают, какие задачи были решены какими участниками. k -ая строка из них описывает, какие задачи были решены участником с номером k . Каждая такая строка содержит T целых чисел, разделенных пробелом. Первое из этих чисел обозначает, решил ли первую задачу участник с номером k . Второе число обозначает то же для второй задачи, и так далее. Эти T чисел могут быть только 0 или 1, где 1 означает, что участник с номером k решил соответствующую задачу, и 0 означает, что он ее не решил.

Формат выходного файла

Ваша программа должна записать в выходной файл одну строку с двумя целыми числами, разделенными одним пробелом. Первое число — количество баллов, которые Бурундуки получили на соревновании ЛКОИ. Второе число — место Бурундуков в финальном протоколе. Место — это целое число от 1 до N включительно, где 1 означает, что участник расположен вверху финального протокола (то есть имеет наибольшее количество баллов), а N означает, что он расположен внизу финального протокола (то есть имеет наименьшее количество баллов).

Пример

lkoi.in	lkoi.out
5 3 2	3 2
0 0 1	
1 1 0	
1 0 0	
1 1 0	
1 1 0	

Задача Н. Простая сортировка

Имя входного файла: qsort.in
 Имя выходного файла: qsort.out
 Ограничение по времени: 1 секунда
 Ограничение по памяти: 64 мегабайт

Дан массив целых чисел. Ваша задача — отсортировать его в порядке неубывания.

Формат входного файла

В первой строке входного файла содержится число N ($1 \leq N \leq 100\,000$) — количество элементов в массиве. Во второй строке находятся N целых чисел, по модулю не превосходящих 10^9 .

Формат выходного файла

В выходной файл надо вывести этот же массив в порядке неубывания, между любыми двумя числами должен стоять ровно один пробел.

Пример

qsort.in	qsort.out
10	1 1 2 2 3 3 4 6 7 8
1 8 2 1 4 7 3 2 3 6	

Задача I. Разброс

Имя входного файла: diff.in
 Имя выходного файла: diff.out
 Ограничение по времени: 3 секунды
 Ограничение по памяти: 64 мегабайт

Дан массив целых чисел. Ваша задача — отсортировать его в порядке неубывания. Разброс между мини-

мальным и максимальным числами в массиве не превышает 107.

Формат входного файла

В первой строке входного файла содержится число N ($1 \leq N \leq 3\,000\,000$) — количество элементов в массиве. Во второй строке находятся N целых чисел, по модулю не превосходящих $2^{31} - 1$.

Формат выходного файла

В выходной файл надо вывести этот же массив в порядке неубывания, между любыми двумя числами должен стоять ровно один пробел.

Пример

diff.in	diff.out
3	3 5 8
5 3 8	

Задача J. Точки и отрезки

Имя входного файла: segments.in
 Имя выходного файла: segments.out
 Ограничение по времени: 1 секунда
 Ограничение по памяти: 64 мегабайт

Дано n отрезков на числовой прямой и m точек на этой же прямой. Для каждой из данных точек определите, скольким отрезкам она принадлежит. Точка x считается принадлежащей отрезку с концами a и b , если выполняется двойное неравенство $\min(a, b) \leq x \leq \max(a, b)$.

Формат входного файла

Первая строка содержит два целых числа n ($1 \leq n \leq 10^5$) — число отрезков и m ($1 \leq m \leq 10^5$) — число точек. В следующих n строках по два целых числа a_i и b_i — координаты концов соответствующего отрезка. В последней строке m целых чисел — координаты точек. Все числа во входном файле не превосходят по модулю 10^9 .

Формат выходного файла

В выходной файл выведите m чисел — для каждой точки количество отрезков, в которых она содержится.

Примеры

segments.in	segments.out
2 2	1 0
0 5	
7 10	
1 6	