

## Задача А. Упаковка символов

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

Билл пытается компактно представить последовательности прописных символов от  $A$  до  $Z$  с помощью упаковки повторяющихся подпоследовательностей внутри них. Например, один из способов представить последовательность  $AAAAAAAAAABABABCCD$  - это  $10(A)2(BA)B2(C)D$ . Он формально определяет сжатые последовательности символов и правила перевода их в несжатый вид следующим образом: Последовательность, содержащая один символ от  $A$  до  $Z$ , является упакованной. Распаковка этой последовательности даёт ту же последовательность из одного символа, повторённую  $x$  раз. Следуя этим правилам, легко распаковать любую заданную упакованную последовательность. Однако Биллу более интересен обратный переход. Он хочет упаковать заданную последовательность так, чтобы результирующая сжатая последовательность содержала наименьшее возможное число символов. Ограничения: длина исходной последовательности от 1 до 100.

### Формат входных данных

В первой строке находится последовательность символов от  $A$  до  $Z$ .

### Формат выходных данных

В единственной строке выводится упакованная последовательность наименьшей длины, которая распаковывается в заданную последовательность. Если таких последовательностей несколько, выведите лексикографически минимальную.

### Примеры

стандартный ввод	стандартный вывод
AAAAAAAAAABABABCCD	9(A)3(AB)CCD
NEERCYESYESYESNEERCYESYESYES	2(NEERC3(YES))
A	A

## Задача В. Покраска забора

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 0.7 секунд  
Ограничение по памяти: 256 мегабайт

256 мегабайт

Мэр города Многоярославца решил построить перед своим домом забор из  $n$  деревянных досок и нанять лучшего маляра города для его покраски. Поскольку забор должен стать главной достопримечательностью города, лучший дизайнер города для каждой доски назначил тщательно выбранный цвет, в который она должна быть покрашена.

Для покраски главный маляр решил применить новейшую технологию, специально разработанную им для выполнения этого задания. Покраской забора будет заниматься специальный робот, который за один час может покрасить произвольный отрезок забора (набор соседних досок) в некоторый цвет. Поскольку задание должно быть выполнено как можно быстрее, требуется составить программу для робота, которая позволит достичь требуемой раскраски за минимальное время. Оставить какую-то из досок непокрашенной, естественно, запрещается.

### Формат входных данных

В первой строке входного файла записано число  $n$  ( $1 \leq n \leq 300$ ), где  $n$  количество досок в заборе. Вторая строка содержит строку из  $n$  символов, описывающую требуемую покраску забора. Цвета обозначаются заглавными латинскими буквами.

### Формат выходных данных

В первой строке выходного файла выведите  $m$  - наименьшее возможное время покраски забора в часах. Следующие  $m$  строк должны содержать программу покраски для робота. Каждая строка должна содержать два числа  $l_i$  и  $r_i$ , а также заглавную букву латинского алфавита, задающую цвет  $c_i$  и означает, что робот должен покрасить участок забора с  $l_i$  по  $r_i$ - доску в цвет  $c_i$  (если длина забора  $n$ , должно выполняться  $1 \leq l_i \leq r_i \leq n$ ).

### Примеры

стандартный ввод	стандартный вывод
5 АВВСА	3 1 5 А 2 3 В 4 4 С
2 АА	1 1 2 А

## Задача C. AliKingspress

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	64 мегабайта

Для экстренного пополнения боеприпасов и вооружения агенты «Кингсман» пользуются службой «AliKingspress». Помимо запроса помощи с вооружением, можно также делать другие запросы, однако уже не бесплатно, а за бонусные баллы. Баллы можно получать каждый день, заходя в специальное приложение. В первый день пользователь получает  $a_1$  баллов, во второй —  $a_2$  баллов, ..., в  $n$ -й день —  $a_n$  баллов. После этого, заходя каждый день, пользователь будет все еще получать  $a_n$  бонусов. Если же пропустить один или несколько дней и не заходить в приложение, при следующем заходе начисление бонусов опять начнется с  $a_1$ .

Эггси посчитал, что для выполнения всех дополнительных запросов, которые он хочет, нужно  $x$  бонусов. Так как он перфекционист, лишние бонусы ему не нужны, он хочет накопить их ровно  $x$ , ни больше, ни меньше. Однако сделать это нужно как можно быстрее, потому что долго ждать он не намерен. Задачу нахождения минимального количества дней, требуемого для этого, он поручил вам — своего верному программисту, пока он сам спасает мир. Помогите ему!

### Формат входных данных

В первой строке содержится два числа  $n$  и  $x$  — количество различных бонусов, а также суммарное количество бонусов, которое нужно набрать Эггси ( $1 \leq n \leq 100, 1 \leq x \leq 10^6$ ).

Во второй строке содержится  $n$  чисел  $a_i$  — размеры бонусов в зависимости от количества дней захода в приложение ( $1 \leq a_i \leq 1\,000$ ).

### Формат выходных данных

В единственной строке выведите минимальное количество дней, нужное для получения ровно  $x$  бонусов или  $-1$ , если набрать ровно  $x$  бонусов невозможно.

### Примеры

стандартный ввод	стандартный вывод
4 14 1 2 3 4	5
5 12 1 4 2 6 3	6
3 8 3 4 2	-1

### Замечание

В первом примере Эггси может заходить в приложение 5 дней подряд и получить, соответственно,  $1 + 2 + 3 + 4 + 4 = 14$  бонусов.

Во втором тестовом примере Эггси может зайти в приложение три дня подряд, затем пропустить один день, а затем зайти еще два дня подряд. В результате он получит  $1 + 4 + 2 + 1 + 4 = 12$  бонусов и потратит на это  $3 + 1 + 2 = 6$  дней.

## Задача D. Морти и подпоследовательности

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

Все знают, как Рик и Морти любят путешествовать и влезать в авантюры! И новое путешествие не исключение! Перед тем как отправиться, Рик попросил Морти помочь ему справиться с одной жизненно-важной задачей, без которой путешествию не состояться. Маленький Морти уже попытался справиться, но у него ничего не вышло, именно поэтому он решил обратиться за помощью к вам!

Задача, которую дал ему Рик звучит следующим образом: дан массив  $a$  из  $n$  целых положительных чисел. Для всех целых  $k$ , для которых выполняется неравенство  $1 \leq k \leq n$  нужно определить, сколько какое максимальное число элементов можно оставить, убрав некоторые, так, чтобы оставшийся массив можно было разбить на подотрезки, каждый из которых — возрастающая последовательность, длины не меньше  $k$ .

Последовательность  $a_{i_1}, a_{i_2}, \dots, a_{i_p}$  называется возрастающей подпоследовательностью в массиве  $a$ , если  $a_{i_1} < a_{i_2} < \dots < a_{i_p}$ .

Размер последовательности — количество элементов, которые принадлежат последовательности.

### Формат входных данных

Первая строка входных данных содержит одно целое число  $n$  ( $1 \leq n \leq 300$ ) отвечающее за длину массива. На второй строке содержится массив  $a$  из  $n$  целых чисел,  $1 \leq a_i \leq 10^9$ .

### Формат выходных данных

В единственной строке выведите  $n$  чисел  $b_i$  — максимальное число элементов, которые войдут в непересекающиеся возрастающие подотрезки размера не менее  $i$  путем исключения некоторого (возможно нулевого) числа элементов из исходного массива.

### Примеры

стандартный ввод	стандартный вывод
3 1 2 3	3 3 3
2 1 1	2 0
5 1 4 3 2 9	5 4 3 0 0

### Замечание

Рассмотри третий пример. Для  $k = 1$ , ответ равен 5, так как каждый элемент по отдельности является возрастающей последовательностью. Для  $k = 2$  максимальный ответ достигается путем избавления, например, от числа 4, разбивая оставшийся массив на два отрезка длины 2, которые являются возрастающими последовательностями. Для  $k = 3$  максимальный ответ можно достичь удалив элементы со значениями 4 и 3, в результате получив один отрезок, который является возрастающей последовательностью длины 3.

## Задача E. [B-A'] 3. Планирование учебного года

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

Уже скоро начнется новый учебный год, и Антон задумался о том, как он хочет его провести. Мальчик считает, что все в этой жизни нужно планировать, в том числе и свои оценки, поэтому он заранее составил для себя несколько правил, которым он будет безукоризненно следовать в течение всего учебного года.

Антон считает, что он должен получать только оценки 3, 4 и 5. Более того, ему известно, что за весь учебный год состоится  $n$  уроков математики, за каждый из которых он получит одну из трех возможных оценок. Антон считает, что год будет удачным, если будут выполнены  $m$  условий.

Каждое из  $m$  условий задается тремя целыми числами  $l_i$ ,  $r_i$  и  $x_i$  ( $1 \leq l_i \leq r_i \leq n$ ,  $1 \leq x_i \leq 3$ ) и гласит, что среди оценок, полученных Антоном на  $l_i$ -м,  $(l_i + 1)$ -м, ...,  $r_i$ -м уроках математики должно быть ровно  $x_i$  различных оценок.

Так как Антон любит математику, он тут же решил посчитать, сколько существует способов заработать оценки на каждом из  $n$  уроков таким образом, чтобы все  $m$  условий были выполнены. А теперь решить данную задачу придется и вам.

### Формат входных данных

Первая строка содержит два целых числа  $n$  и  $m$  ( $1 \leq n, m \leq 300$ ) — количество уроков математики, которые состоятся в учебном году, и количество условий, которые должен выполнить Антон.

Каждая из следующих  $m$  строк содержит три целых числа  $l_i$ ,  $r_i$  и  $x_i$  ( $1 \leq l_i \leq r_i \leq n$ ,  $1 \leq x_i \leq 3$ ) — описание  $i$ -го условия, которое должен выполнить Антон.

### Формат выходных данных

Выведите одно целое число — количество способов заработать оценки на каждом из  $n$  уроков таким образом, чтобы все  $m$  условий были выполнены.

Так как ответ может быть достаточно большим, выведите остаток от деления ответа на число  $10^9 + 7$ .

### Примеры

стандартный ввод	стандартный вывод
3 2 1 3 3 1 2 1	0
3 3 2 2 1 1 3 2 2 3 1	6

### Замечание

Рассмотрим первый пример из условия. Всего за год состоится три урока математики, причем Антон должен выполнить два условия:

- Среди оценок, полученных за уроки 1, 2 и 3, должно быть ровно три различных оценки.
- Среди оценок, полученных за уроки 1 и 2, должна быть ровно одна различная оценка.

Нетрудно понять, что эти два условия противоречат друг другу, поэтому количество способов заработать оценки равно нулю.

Во втором примере из условия подходят следующие способы получить оценки:

1. [1, 2, 2]
2. [1, 3, 3]
3. [2, 1, 1]
4. [2, 3, 3]
5. [3, 1, 1]
6. [3, 2, 2]

## Задача F. Максимальная тройка

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

Дано дерево на  $n$  вершинах. Требуется выбрать из них три так, чтобы сумма расстояний между ними была максимальна.

### Формат входных данных

Первая строка каждого теста содержит натуральное число  $n$  — количество вершин в дереве ( $3 \leq n \leq 1\,000\,000$ ). Следующие  $n - 1$  строк содержат по 2 натуральных числа  $v, u$  и описывают ребро дерева, соединяющее две вершины  $v$  и  $u$  ( $1 \leq v, u \leq n$ ).

### Формат выходных данных

Выведите единственное число — максимальную сумму расстояний.

### Система оценки

Решения, правильно работающие на тестах, в которых  $n \leq 50$ , будут оцениваться в 25 баллов.

Решения, правильно работающие на тестах, в которых  $n \leq 500$ , будут оцениваться в 50 баллов.

Решения, правильно работающие на тестах, в которых  $n \leq 5000$ , будут оцениваться в 75 баллов.

Решения, правильно работающие на тестах, в которых  $n \leq 10^6$ , будут оцениваться в 100 баллов.

### Примеры

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

## Задача G. Логическое дерево

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

Рассмотрим разновидность двоичного дерева, которую мы назовем логическим деревом. В этом дереве каждый уровень полностью заполнен, за исключением, возможно, последнего (самого глубокого) уровня. При этом все вершины последнего уровня находятся максимально слева. Дополнительно, каждая вершина дерева имеет ноль или двоих детей.

Каждая вершина дерева имеет связанное с ней логическое значение (1 или 0). Кроме этого, каждая внутренняя вершина имеет связанную с ней логическую операцию («И» или «ИЛИ»). Значение вершины с операцией «И» — это логическое «И» значений её детей. Аналогично, значение вершины с операцией «ИЛИ» — это логическое «ИЛИ» значений её детей. Значения всех листьев задаются во входном файле, поэтому значения всех вершин дерева могут быть найдены.

Наибольший интерес для нас представляет корень дерева. Мы хотим, чтобы он имел заданное логическое значение  $v$ , которое может отличаться от текущего. К счастью, мы можем изменять логические операции некоторых внутренних вершин (заменить «И» на «ИЛИ» и наоборот).

Дано описание логического дерева и набор вершин, операции в которых могут быть изменены. Найдите наименьшее количество вершин, которые следует изменить, чтобы корень дерева принял заданное значение  $v$ . Если это невозможно, то выведите строку «IMPOSSIBLE» (без кавычек).

### Формат входных данных

В первой строке входного файла находятся два числа  $n$  и  $v$  ( $1 \leq n \leq 10\,000$ ,  $0 \leq v \leq 1$ ) — количество вершин в дереве и требуемое значение в корне соответственно. Поскольку все вершины имеют ноль или двоих детей, то  $n$  нечётно. Следующие  $n$  строк описывают вершины дерева. Вершины нумеруются от 1 до  $n$ .

Первые  $(n-1)/2$  строк описывают внутренние вершины. Каждая из них содержит два числа —  $g$  и  $s$ , которые принимают значение либо 0, либо 1. Если  $g = 1$ , то вершина представляет логическую операцию «И», иначе она представляет логическую операцию «ИЛИ». Если  $s = 1$ , то операция в вершине может быть изменена, иначе нет. Внутренняя вершина с номером  $i$  имеет детей  $2i$  и  $2i+1$ .

Следующие  $(n+1)/2$  строк описывают листья. Каждая строка содержит одно число 0 или 1 — значение листа.

### Формат выходных данных

В выходной файл выведите ответ на задачу.



## Примеры

стандартный ввод	стандартный вывод
9 1 1 0 1 1 1 1 0 0 1 0 1 0 1	1
5 0 1 1 0 0 1 1 0	IMPOSSIBLE

## Задача Н. Миллиардная Функция Васи

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

Вася — начинающий математик — решил сделать вклад в развитие этой науки и прославиться на весь мир. Но как это сделать, когда самые интересные факты, типа теоремы Пифагора, давно уже доказаны? Правильно! Придумать что-то свое, оригинальное. Вот юный математик и придумал Теорию Функций Васи, посвященную изучению поведения этих самых функций. Функции Васи (ФВ) устроены довольно просто: значением  $N$ -й ФВ в точке  $S$  будет количество чисел от 1 до  $N$ , имеющих сумму цифр  $S$ . Вам, как крутым программистам, Вася поручил найти значения миллиардной ФВ (то есть ФВ с  $N = 10^9$ ), так как сам он с такой задачей не справится. А Вам слабо?

### Формат входных данных

В единственной строке записано целое число  $S$  ( $1 \leq S \leq 81$ ).

### Формат выходных данных

Выведите одно число — значение миллиардной Функции Васи в точке  $S$ .

### Пример

стандартный ввод	стандартный вывод
1	10

## Задача I. Интересные числа

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

Софья считает число интересным, если его цифры идут в неубывающем порядке. Например, числа 123, 1111 или 888999 – интересные.

Софья заинтересовалась, сколько существует интересных положительных чисел, лежащих в диапазоне от  $L$  до  $R$  включительно. Это число может оказаться довольно большим для больших  $L$  и  $R$ , поэтому Софья хочет найти остаток от деления этого числа на  $10^9 + 7$ .

Требуется написать программу, которая по заданным  $L$  и  $R$  определяет количество интересных чисел, лежащих в диапазоне от  $L$  до  $R$  включительно, и выводит остаток от деления этого числа на  $10^9 + 7$ .

### Формат входных данных

Входной файл содержит две строки. Первая строка содержит число  $L$ , вторая строка содержит число  $R$  ( $1 \leq L \leq R \leq 10^{100}$ ).

### Формат выходных данных

Выходной файл должен одно целое число — остаток от деления количества интересных чисел, лежащих в диапазоне от  $L$  до  $R$  включительно, на  $10^9 + 7$ .

### Пример

стандартный ввод	стандартный вывод
1 100	54

## Задача J. Плавные числа

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

Назовём натуральное число плавным, если значения соседних цифр отличаются не более, чем на

1. Определите количество  $N$ -значных плавных чисел. Запись числа не может начинаться с цифры 0.

### Формат входных данных

Программа получает на вход одно целое число  $N$  ( $1 \leq N \leq 20$ ).

### Формат выходных данных

Программа должна вывести одно целое число — искомое количество плавных чисел.

### Пример

стандартный ввод	стандартный вывод
2	26

## Задача К. Трипростые числа

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

Будем называть натуральное число трипростым, если в нем любые подряд идущие 3 цифры образуют трехзначное простое число. Требуется по данному  $N$  найти количество  $N$ -значных трипростых чисел.

### Формат входных данных

На вход подаётся одно натуральное число  $N$ : ( $3 \leq N \leq 10^4$ ).

### Формат выходных данных

Ответ должен содержать количество  $N$ -значных трипростых чисел, которое следует вывести по модулю  $10^9 + 9$ .

### Примеры

стандартный ввод	стандартный вывод
3	143
4	204
4793	851557205

## Задача L. Black Friday

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

Сегодня — черная пятница. В честь этого Миша решил закупить оборудование для своей майнинг фермы. В магазине действует удивительная акция, если купить видеокарту номер  $i$  в количестве от  $l_i$  до  $r_i$ , включительно ( $1.4 \cdot l_i \leq r_i$ ), то их стоимость будет сильно ниже рыночной. Миша может унести с собой не больше  $s$  видеокарт. Из-за такой щедрости магазина он решил купить как можно больше видеокарт по этой акции. Решить такую задачу Мише не представляется возможным, поэтому он просит вас о помощи. Для каждой видеокарты определите, сколько штук надо купить, либо не покупать вообще, чтобы все видеокарты были куплены по акции, при этом их количество было **максимально возможным** числом не большим  $s$ .

### Формат входных данных

Первая строка входных данных содержит два целых числа  $n$  и  $s$  ( $1 \leq n \leq 10^5$ ,  $1 \leq s \leq 10^{13}$ ) — количество различных видеокарт и максимальное число видеокарт, которое Миша может унести с собой.

Следующие  $n$  строк содержат по два целых числа  $l_i$  и  $r_i$  ( $1 \leq l_i \leq 10^{13}$ ,  $1.4 \cdot l_i \leq r_i \leq 10^{13}$ ) — минимальное и максимальное количество видеокарт типа  $i$ , которое можно купить по скидке.

### Формат выходных данных

В первой строке выведите единственное целое число  $w$  — максимальное количество видеокарт, которое можно купить по скидке ( $0 \leq w \leq s$ ). Во второй строке выведите через пробел  $n$  целых чисел  $x_i$ .  $i$ -е число равно количеству видеокарт типа  $i$ , которое необходимо купить. Заметим, что либо  $x_i = 0$ , либо  $l_i \leq x_i \leq r_i$ , а также что сумма  $x_i$  равна  $w$ . Если существует несколько решений, максимизирующих  $w$ , выведите любое.

### Система оценки

Подзадача	Баллы	Дополнительные ограничения
1	10	$n \leq 20$
2	14	$n \leq 500$ , $s \leq 2 \cdot 10^5$
3	16	$n \leq 500$
4	29	$n \leq 5000$
5	31	нет дополнительных ограничений

### Пример

стандартный ввод	стандартный вывод
3 20	19
1 2	2 17 0
10 17	
11 16	

### Пояснение к примеру

В примере из условия выгодно купить видеокарты первого и второго типа. Заметим, что нельзя одновременно купить видеокарты второго и третьего типа, так как тогда нужно будет купить хотя бы  $10 + 11 = 21$  штуку, что больше 20.

## Задача М. Высокородная дама

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

Кырт был основным источником дохода для всего Сарка, выводя его экономику на второе место в Галактике, разумеется, после Трантора, под контролем которого была не одна сотня тысяч других планет. Его физические характеристики поражали: он мог сиять на солнце любым цветом или всем спектром сразу, был невосприимчив ко многим химическим веществам, а по прочности ему уступали любые сорта стали. Однако ученым никак не удавалось понять, чем вызваны столь необычные свойства.

Сэмия Файфская, дочь великого сквайра Файфа, хоть и была высокородной дамой, тоже интересовалась этим вопросом. Другие аристократки относились пренебрежительно к увлечению наукой, однако Сэмию это ничуть не волновало. Вот уже пять лет она подробно изучала свойства кырта, планету, где он рос, и людей, которые его выращивали. Она настояла на том, чтобы полететь на Флорину, хотела провести несколько месяцев на полях и на фабрике. Но новости о нападении на патрульных уже дошли до Сарка, и отец приказал Сэмии вернуться домой ради её же безопасности.

Проблема заключалась в том, что всё оборудование, образцы кырта и вещи Сэмии вряд ли бы поместились на небольшой корабль. Поэтому она решила взять наиболее ценные вещи, которые можно было увезти с собой. Для каждого из  $n$  предметов Сэмия оценила его вес  $w_i$ , а так же ценность  $v_i$ . Выбрать вещи она решила следующим образом: расставить предметы в порядке невозрастания весов, а при равенстве весов — невозрастания ценностей. Затем Сэмия просматривает предметы в этом порядке, и берет очередной предмет, если его вес с учетом веса уже взятых предметов не превосходит грузоподъемности корабля.

Но найти корабль, достойный такой высокородной дамы на Флорине непросто, и капитан Рейсти еще не уверен, на каком из кораблей они полетят. Поэтому  $q$  раз происходит следующее событие: капитан принимает решение (к сожалению, не окончательное), выбирает корабль и сообщает Сэмии его грузоподъемность  $s_i$ . Вам предстоит для каждого из кораблей узнать, если Сэмия будет выбирать предметы описанным образом, какова будет суммарная ценность выбранных предметов.

### Формат входных данных

В первой строке содержится число  $n$  — количество предметов, которые Сэмия хочет взять с собой ( $1 \leq n \leq 2 \cdot 10^5$ ).

В следующих двух строках содержатся по  $n$  целых чисел — соответственно веса  $w_i$  и ценности  $v_i$   $i$ -го предмета ( $1 \leq w_i \leq 5 \cdot 10^{12}$ ,  $1 \leq v_i \leq 10^9$ ).

Следующая строка содержит число запросов  $q$  — количество кораблей, которые рассмотрит для полета капитан ( $1 \leq q \leq 2 \cdot 10^5$ ).

Последующие  $q$  строк содержат по одному целому числу  $s_i$  — грузоподъемность  $i$ -го корабля ( $1 \leq s_i \leq 10^{18}$ ).

### Формат выходных данных

Выведите  $q$  строк, в  $i$ -й строке ценность предметов, которые могла бы увезти Сэмия на  $i$ -м корабле.

## Пример

стандартный ввод	стандартный вывод
5	12
5 5 2 4 10	12
7 6 2 5 12	14
5	19
9	25
11	
13	
15	
21	

## Замечание

В данном примере Сэмия расставит предметы в таком порядке: 5, 1, 2, 4, 3.

Если грузоподъемность корабля равна 9, Сэмия сможет взять на борт 1 и 4 предметы суммарной ценностью  $7 + 5 = 12$ .

На корабль с грузоподъемностью 11 Сэмия возьмет только 5 предмет, который имеет ценность 12.

Если грузоподъемность корабля увеличится до 13, то вместе с 5 предметом Сэмия возьмет также 3 предмет, получив суммарную ценность  $12 + 2 = 14$ .

На корабль с грузоподъемностью 15 удастся взять 5 и 1 предмеры, имеющие ценность  $12 + 7 = 19$ .

Наконец, если грузоподъемность корабля равна 21, Сэмия возьмет с собой 5, 1 и 2 предметы, имеющие ценность  $12 + 7 + 6 = 25$ .



## Задача N. Большой рюкзак

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

У вас есть  $N$  предметов с весами  $w_1, w_2, \dots, w_N$ . Требуется проверить, можно ли выбрать некоторые предметы, суммарный вес которых равен  $W$ .

### Формат входных данных

В первой строке записаны два целых числа  $N$  и  $W$  ( $1 \leq N \leq 2500$ ,  $1 \leq W \leq 6250000$ ).

Во второй строке через пробел записаны  $N$  целых чисел  $w_1, w_2, \dots, w_N$  ( $1 \leq w_i \leq 2500$ ) — веса предметов.

### Формат выходных данных

В единственной строке выведите «YES» (без кавычек), если, используя данные предметы, можно набрать вес  $W$ , либо «NO» в противном случае.

### Примеры

стандартный ввод	стандартный вывод
5 10 1 2 3 4 5	YES
2 10 4 5	NO

## Задача О. Большой рюкзак, но теперь с восстановлением!

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

У вас есть  $N$  предметов с весами  $w_1, w_2, \dots, w_N$ . Требуется проверить, можно ли выбрать некоторые предметы, суммарный вес которых равен  $W$ .

### Формат входных данных

В первой строке записаны два целых числа  $N$  и  $W$  ( $1 \leq N \leq 2\,500$ ,  $1 \leq W \leq 6\,250\,000$ ).

Во второй строке через пробел записаны  $N$  целых чисел  $w_1, w_2, \dots, w_N$  ( $1 \leq w_i \leq 2\,500$ ) — веса предметов.

### Формат выходных данных

В первой строке выведите «YES» (без кавычек), если, используя данные предметы, можно набрать вес  $W$ , либо «NO» в противном случае.

В случае, если набрать вес  $W$  возможно, восстановите ответ.

Во второй строке выведите одно целое число  $K$  ( $1 \leq K \leq N$ ) — количество выбранных предметов.

В третьей строке выведите через пробел  $K$  различных целых чисел  $i_1, i_2, \dots, i_K$  ( $1 \leq i_j \leq N$ ) — номера выбранных предметов.

### Примеры

стандартный ввод	стандартный вывод
5 10 1 2 3 4 5	YES 4 1 2 3 4
2 10 4 5	NO

## Задача Р. Гирьки: кучки одного размера

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

Дан набор гирек массой  $m_1, \dots, m_N$ . Разделите этот набор на две кучки равной массы, содержащие равное число гирек.

### Формат входных данных

Первая строка входных данных содержит натуральное число  $N$ , не превышающее 100.

В следующей строке через пробел записаны  $N$  натуральных чисел  $m_i$ , не превышающих 100.

### Формат выходных данных

Необходимо вывести в первой строчке номера гирек (числа от 1 до  $N$ ), входящие в первую кучку, во второй строчке — номера гирек во второй кучке.

Если задача не имеет решения, выведите одно число -1.

### Пример

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

## Задача Q. Гирьки: три кучки

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

Дан набор гирек массой  $m_1, \dots, m_N$ . Можно ли их разложить на три кучки равной массы?

### Формат входных данных

Первая строка входных данных содержит натуральное число  $N$ , не превышающее 60.

Во второй строке через пробел записаны  $N$  натуральных чисел  $m_i$ , не превышающих 60.

### Формат выходных данных

Выведите три строки, описывающие наборы гирек, либо число  $-1$ , если решение не существует.

В каждой строке сначала выведите количество гирек в соответствующем наборе, а затем через пробел номера гирек в наборе.

### Пример

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