

## Задача А. Плитки

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	512 мегабайт

Сегодня у Ивана произошла неприятность. Он вышел во двор и обнаружил, что некоторые тротуарные плитки треснули. Взяв необходимые инструменты, наш герой принялся измерять размеры поврежденной площади. После измерений Иван получил размер площади  $S$ , которую нужно теперь ему покрыть плиткой.

К счастью, наш герой обнаружил  $N$ -ое количество треугольных плиток у себя в сарае. Стоит отметить, что каждая такая плитка имеет свои определенные длины сторон, их то как раз и принялся измерять Иван. Наш герой любил прогуливать уроки геометрии в школе, поэтому просит вас ответить на вопрос: **Хватит ли Ивану данного количества плиток, чтобы покрыть искомую площадь  $S$ ?**

Так как плитки тяжелые, Иван задаётся ещё одним вопросом: **Какое наименьшее количество плиток ему нужно забрать, чтобы покрыть искомую площадь  $S$ ?**

При решении данной задачи не нужно учитывать форму плиток.

### Формат входных данных

Первая строка входных данных содержит два целых числа  $N$  ( $1 \leq N \leq 100$ ) и  $S$  ( $1 \leq S \leq 10^4$ ) — количество треугольных плиток в сарае и искомая площадь для покрытия.

$i$ -я из следующих  $N$  строк содержит три целых числа  $a, b, c$  ( $1 \leq a, b, c \leq 10^3$ ) — длины сторон треугольника.

Гарантируется, что треугольники корректные (невырожденные).

### Формат выходных данных

Выведите «YES», если Ивану хватит данного количества треугольников, чтобы покрыть искомую площадь  $S$  и выведите с новой строки минимальное количество треугольников, с помощью которых можно покрыть искомую площадь  $S$ .

Иначе выведите «NO» и с новой строки выведите площадь, которой не хватает для покрытия искомой площади  $S$ .

Ваш ответ будет считаться правильным, если его абсолютная или относительная погрешность не превосходит  $10^{-6}$ .

### Система оценки

Подзадачи	Ограничения	Баллы	Необходимые подзадачи	Тип проверки
1	$1 \leq N \leq 100$ $1 \leq S \leq 10^4$ $1 \leq a, b, c \leq 10^3$	100	—	Каждый тест

## Примеры

стандартный ввод	стандартный вывод
5 37 2 3 4 8 3 7 12 9 8 4 5 2 7 4 4	YES 2
5 60 2 3 4 8 3 7 12 9 8 4 5 2 7 4 4	NO 0.126434

## Замечание

В первом примере мы можем взять две плитки, суммарной площади которых будет вполне достаточно, чтобы покрыть искомую площадь  $S=37$ .

## Задача В. Кепарик и Кибер-скарабей

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

Одним теплым весенним утром два друга Кепарик и Второй решили проверить, кто из них лучший наездник. Состязаться решили в скачках на Кибер-скарабей в Тарвильтордовых горах. Каждый из товарищей может стартовать с любой вершины и двигаться вперед. Победителем считается участник, преодолевший большее количество гор, отталкиваясь от каждой последующей вершины. (Вершина считается преодоленной, если участник смог оказаться на ней после прыжка).

Код, написанный «прямыми» руками разработчика, заложенный в Кибер-скарабей, не позволяет ему прыгать выше высоты, с которой они начали скачок, зато Кибер-скарабей имеет трансильфузный скачок через киберпространство, поэтому способен преодолевать большое расстояние, не теряя высоты и облетая горы на своем пути.

Кепарик раздобыл у товарищей гномов магическую карту гор, которая показывает путь по загаданному количеству вершин, и решил схитрить. Карта может засветить путь лишь один раз, поэтому он просит вас помочь ему и подсказать, какое максимальное количество вершин он может преодолеть, чтобы карта его засветила.

### Формат входных данных

Первая строка содержит одно целое число  $N$  ( $2 \leq N \leq 10^6$ ) — количество вершин Тарвильтордовых гор.

Вторая строка содержит  $N$  целых чисел  $A_i$  — высота  $i$ -й вершины ( $1 \leq A_i \leq 727$ ).

### Формат выходных данных

Выведите одно число — максимальное количество преодоленных вершин.

### Система оценки

Подзадачи	Ограничения	Баллы	Необходимые подзадачи	Тип проверки
1	$N \leq 10^3$	15	—	Каждый тест
2	$N \leq 10^4$	25	—	Каждый тест
3	$N \leq 10^6$	60	—	Полная

### Примеры

стандартный ввод	стандартный вывод
9 15 27 17 15 6 4 1 1 13	6
15 9 5 4 1 3 5 2 6 5 4 2 5 7 8 15	5

## Задача С. Сокращение слов

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

Когда-то в давние времена люди умели писать только 0 и 1. С помощью длинных слов из нулей и единиц они общались между собой. Но однажды царские библиотеки переполнились объёмными отчётами, и царь решил: пришла пора реформ. Он издал указ сократить слова по следующим правилам:

1. Пока в слове присутствует подслово «10», заменить его на «01». Если подслов «10» нет, перейти к шагу 2.
2. Пока в слове присутствует подслово «00», удалить его вхождение. Если подслов «00» нет, перейти к шагу 3.
3. Пока в слове присутствует подслово «11», удалить его вхождение. Если подслов «11» нет, завершить процесс сокращения слова.

На вас возложена задача написать программу, которая исправит старые тексты в соответствии с указанными правилами.

Подсловом слова  $S$  является любое слово, полученное из  $S$  путём удаления нескольких (возможно, ни одного) символов из начала и нескольких (возможно, ни одного) символов с конца.

Обратите внимание, что после завершения процесса сокращения может получиться **пустое слово**. В этом случае вместо него следует вывести  $-1$ .

### Формат входных данных

Первая строка содержит целое число  $N$  ( $1 \leq N \leq 2 \cdot 10^5$ ) — количество слов в тексте, который необходимо исправить.

В следующей строке записаны  $N$  слов, разделённых пробелом и состоящих только из символов «0» и «1». Сумма длин слов не превосходит  $2 \cdot 10^5$ . Гарантируется, что все слова непустые.

### Формат выходных данных

Выведите единственную строку — слова текста в том же порядке, преобразованные по указанным правилам и разделённые пробелом. Если после преобразования получено **пустое слово**, следует вывести  $-1$  вместо него.

### Система оценки

Подзадачи	Ограничения	Баллы	Необходимые подзадачи	Тип проверки
1	$1 \leq N \leq 10^5$	90	—	Каждый тест
2	$N = 2 \cdot 10^5$	10	1	Полная

### Пример

стандартный ввод	стандартный вывод
5 01010 11111 0010 11010001 0	0 1 01 -1 0

## Задача D. Круглый магазин

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

После тяжелого рабочего дня, по дороге домой Даниил решил зайти в недавно открывшийся магазин. Ещё утром он составил список из  $M$  продуктов, которые ему необходимо купить. Он уверен, что все они есть в наличии. Однако, когда он зашёл в магазин, перед ним появилась проблема. Оказалось, что все полки с продуктами расположены по кругу. При входе он увидел каталог из  $N$  неповторяющихся товаров. Продукты на полках расположены в том же порядке, что и в каталоге, т.е. за первым товаром идёт второй, за вторым — третий, за  $N$ -м — первый. Даниил может ходить по магазину как по часовой стрелке, так и против часовой стрелки. Для того, чтобы добраться от полки с текущим товаром к полкам со следующим или с предыдущим, Даниил тратит 1 секунду (товар он забирает мгновенно). Наш герой боится что-то забыть, поэтому он идёт строго по своему списку и не покупает следующий товар, пока не купит предыдущий.

Определите минимальное время, которое необходимо Даниилу, чтобы купить все продукты. Стоит учитывать, что он оказался перед полкой с первым товаром, когда зашёл в магазин, а выйти он сможет мгновенно после покупки последней позиции из своего списка.

### Формат входных данных

Первая строка входных данных содержит целое число  $N$  ( $1 \leq N \leq 10^5$ ) — количество продуктов в магазине.

Далее следует  $N$  неповторяющихся строк, состоящих из строчных латинских букв — названия товаров в магазине. Длины строк не превышают 50 символов.

Следующая строка содержит целое число  $M$  ( $1 \leq M \leq N$ ) — количество продуктов в списке Даниила.

Далее следует  $M$  неповторяющихся строк, состоящих из строчных латинских букв — названия товаров из списка Даниила. Длины строк не превышают 50 символов.

### Формат выходных данных

Выведите единственное целое число — минимальное количество времени, которое необходимо Даниилу, чтобы купить все продукты.

### Система оценки

Подзадачи	Ограничения	Баллы	Необходимые подзадачи	Тип проверки
1	$N \leq 10, M \leq 10$	5	—	Полная
2	$N \leq 10^2, M \leq 10^2$	10	1	Полная
3	$N \leq 10^5, M \leq 10$	15	2	Полная
4	$N \leq 10^5, M \leq 10^3$	25	3	Полная
5	$N \leq 10^5, M \leq 10^5$	45	4	Каждый тест

## Примеры

стандартный ввод	стандартный вывод
7 milk bread apple meat cheese lemon eggs 4 apple eggs lemon milk	8
2 abc gh 1 abc	0

## Задача Е. Составь больше

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

Два одноклассника Василий и Прохор играют с набором дощечек. На каждой дощечке нарисован один из трех символов:  $-$ ,  $0$ ,  $+$ . Парни любили все сравнивать, поэтому договорились, что  $- < 0 < +$ . Василий выступает в роли раскладчика дощечек в ряд, а Прохор должен посчитать сколько есть способов переставить дощечки так, чтобы новая последовательность символов была лексикографически больше чем исходная (Прохор может сколько угодно раз перемещать местами пластинки). У Василия плохо со счетом, поэтому он просит вас помочь ему.

### Формат входных данных

Первая строка содержит одно целое число  $N$  ( $1 \leq N \leq 50$ ) — количество входных строчек.

Следующие  $N$  строк содержат последовательность из символов:  $-$ ,  $0$ ,  $+$  длиной  $1 < S < 15$ .

### Формат выходных данных

Выведите количество перестановок, которые лексикографически больше для каждой строки.

### Система оценки

Подзадачи	Кол-во строк в тесте	Баллы	Необходимые подзадачи	Тип проверки
1	$N \leq 10$	15	—	Каждый тест
2	$N \leq 25$	25	—	Каждый тест
3	$N \leq 50$	60	—	Полная

### Пример

стандартный ввод	стандартный вывод
5	7
--0	1
-0-	7
--0	0
+++	10
-+0+	

### Замечание

К примеру: Дана последовательность «- 0 -». Единственная перестановка, которая лексикографически больше исходной, это «0 - -». Еще один пример: Дана строка «0 - +». Из нее мы можем перестановками сделать еще 5 последовательностей. «0 + -», «+ 0 -», «+ - 0», «- 0 +», «- + 0». Среди них только 1, 2 и 3 будут лексикографически больше исходной.