

Наибольший подпалиндром

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 4 секунды
Ограничение по памяти: 256 мегабайт

Дана строка длины N . Требуется найти наибольшую подстроку, являющуюся палиндромом.

Формат входных данных

В первой и единственной строке входного файла дана последовательность заглавных и строчных букв английского алфавита. Длина последовательности $1 \leq N \leq 1000$.

Формат выходных данных

В первой строке выведите длину наибольшего подпалиндрома. Во второй строке выведите подпалиндром. Если наибольших подпалиндромов несколько, выведите первый из них.

Примеры

стандартный ввод	стандартный вывод
abaca	3 aba

Электронная почта

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

Дед Мороз решил поздравить детей с Новым годом по электронной почте. Чтобы быстро составить большую базу адресов для массовой рассылки, Дед Мороз решил написать программу, сканирующую произвольный текст и находящую в нём адреса электронной почты. Помогите ему.

Напомним, что адрес электронной почты записывается в формате `имя_пользователя@домен`, где `имя_пользователя` и `домен` — непустые строки, состоящие из строчных и заглавных латинских букв и точек. Имя пользователя и домен не могут начинаться с точки, заканчиваться точкой и содержать более одной точки подряд.

Формат входных данных

Входные данные содержат текст, который должна сканировать программа Деда Мороза. Текст состоит из заглавных и строчных латинских букв, пробелов, точек, знаков `@` и переводов строки. Размер текста не превышает 1Кб.

Формат выходных данных

Выведите количество адресов электронной почты, которые содержатся в этом тексте.

Примеры

<code>email.in</code>	<code>email.out</code>
<code>lksh@lksh.ru is a valid address. e.e@e.e@e.e and @example.com are not.</code>	<code>1</code>

Числа

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

Все числа от 1 до N включительно записали подряд без пробелов. После этого некоторые цифры стёрли. Известно, какие цифры остались. Найдите минимальное N , для которого это могло произойти.

Формат входных данных

В единственной строке записано не менее одной и не более 2500 цифр, оставшихся после стирания.

Формат выходных данных

Выведите единственное целое число — минимальное N , для которого ситуация возможна.

Примеры

стандартный ввод	стандартный вывод
1234	4
234092	20
999909	49
82340329923	43

Выходные дни

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

В некотором государстве год состоит из N дней, причём неделя состоит из w подряд идущих рабочих дней. Профсоюзы требуют, чтобы после w рабочих дней было добавлено h выходных дней, тем самым неделя будет состоять из $w + h$ дней — сначала w рабочих дней, затем h выходных, снова w рабочих дней, затем h выходных и т.д. Последняя неделя может быть неполной (если N не делится на $w + h$), тогда в неполной неделе сначала идут рабочие дни (не более w), затем выходные.

Олигархи согласны на такую реформу календаря, но настаивают, чтобы общее число рабочих дней в году было не менее M . Определите, какое наибольшее число выходных дней в каждой неделе можно добавить в календарь так, чтобы выполнить требования олигархов.

Формат входных данных

Программа получает на вход три целых числа N , M , w — записанных в одной строке через пробел. N — количество дней в году, M — минимальное общее число рабочих дней в году, w — число рабочих дней в одной неделе. $1 \leq N \leq 10^{18}$, $1 \leq M \leq N$, $1 \leq w < M$.

Формат выходных данных

Программа должна вывести одно целое число — максимальное число выходных дней, которое можно добавить в неделю так, что общее число рабочих дней в году будет не менее M .

Примеры

<code>holidays.in</code>	<code>holidays.out</code>
100 70 8	3
10 3 2	7

Королевский слон

Имя входного файла: `c3po.in`
Имя выходного файла: `c3po.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Знали ли вы, что на Татуине играют в некоторое подобие человеческих шахмат? На песке рисуется клетчатое поле размера $h \times w$, где h — количество клеток с севера на юг, а w — с запада на восток. На клетке в юго-западном углу стоит дроид С-ЗРО в режиме «король». Два игрока по очереди перемещают дроида в соответствии с его режимом, а в конце хода выбирают, в каком режиме — «король» или «слон» — будет работать дроид на следующем ходу. Дроид всегда может ходить только на север, восток или северо-восток, при этом в режиме «король» ходит как шахматный король, а в режиме «слон» — как шахматный слон. Побеждает тот, кто поставит С-ЗРО в северо-восточный угол, при этом до завершения игры запрещено менять режим так, чтобы у игрока не было возможности сделать ход.

Определите, кто победит при правильной игре.

Формат входных данных

Во входном файле записаны два целых числа h и w — размеры поля ($2 \leq h, w \leq 50$).

Формат выходных данных

В выходной файл выведите «FIRST», если выигрывает первый игрок, и «SECOND», если выигрывает второй.

Примеры

<code>c3po.in</code>	<code>c3po.out</code>
2 5	FIRST
4 4	SECOND

Тест на тупость: 2

Имя входного файла: `sum2.in`
Имя выходного файла: `sum2.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

У Александра есть n палочек длинами a_1, \dots, a_n . И ещё n палочек длинами b_1, \dots, b_n . Ваша задача — определить, может ли Саша из одной палочки первого набора и одной палочки второго набора составить палочку длиной s .

Формат входных данных

В первой строке записаны числа n и s — количество чисел в наборах и требуемая длина. Во второй и третьей строке записано по n натуральных чисел: наборы a и b . Наборы отсортированы по неубыванию. $n \leq 100\,000$. Все числа — натуральные, ни одно не превышает 10^9 .

Формат выходных данных

Выведите пару чисел i, j , если палочки a_i и b_j складываются в палочку длины s . Если такой пары палочек нет, выведите одну пару нулей.

Примеры

<code>sum2.in</code>	<code>sum2.out</code>
3 7 2 3 9 1 5 7	1 2

Камни

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

На столе лежат N камней. За ход игрок может взять:

- 1 или 2 камня, если N делится на 3;
- 1 или 3, если N при делении на 3 дает остаток один;
- 1, 2 или 3, если N при делении на 3 дает остаток два.

Каждый ход можно сделать при наличии достаточного количества камней. Проигрывает тот, кто хода сделать не может.

Формат входных данных

Вводится целое число $0 < N \leq 100$.

Формат выходных данных

Выведите 1 или 2 – номер игрока, который выиграет при правильной игре.

Примеры

<code>stones.in</code>	<code>stones.out</code>
3	2
5	1