

## Общая информация

Всего в соревновании шесть задач. Во всех задачах входные данные следует читать из стандартного потока ввода («с клавиатуры»), а выводить следует в стандартный поток вывода («на экран»), причём только то, что требуется в задаче.

Задачи можно решать в любом порядке. В каждой задаче работа решения на каждом тесте оценивается отдельно. Чтобы пройти тест, программа должна не только вывести правильный ответ, но и успеть сделать это за 2 секунды, используя не более 512 мегабайт памяти. Максимальное количество баллов за каждую задачу равно 100.

Решения, которые не работают на примерах из условия, баллы получить не могут.

## Задача А. Повтор

Зизи — чат-бот. У неё есть одна особенность: каждый раз, когда ей нужно написать двузначное число, она пишет его два раза подряд, и получается четырёхзначное число. Например, если спросить её: «сколько будет пятью пять?», она ответит не «25», а «2525».

Дано четырёхзначное число, которое из-за своей особенности написала Зизи. Какое двузначное число она имела в виду?

### Формат входных данных

В первой строке записано четырёхзначное целое число  $n$ , которое написала Зизи.

### Формат выходных данных

В первой строке выведите одно двузначное целое число: то, которое имела в виду Зизи.

### Примеры

<i>стандартный ввод</i>	<i>стандартный вывод</i>
2525	25
7777	77

## Задача В. Гадание

Мари — электронная гадалка. Для общения с ней нужно сперва загадать вопрос, предполагающий ответ «да» или «нет». После этого нужно, держа в голове этот вопрос, послать Мари немного криптовалюты.

В ответ Мари пришлёт обратно строку из маленьких английских букв. Далее следует внимательно прочитать эту строку слева направо, ища нужные слова: три буквы «yes» подряд или две буквы «no» подряд. То из этих слов, которое встретилось раньше (левее) — это и есть ответ на вопрос. Если же не встретилось ни одно из этих слов — значит, вопрос сложный, и посланной криптовалютой оказалось недостаточно для ответа.

Сыщица Софи подозревает, что гадание Мари — надувательство. Чтобы доказать это или опровергнуть, ей нужна программа, которая по заданной строке выясняет, какой получен ответ: «yes», «no» или никакого. Напишите для неё такую программу.

### Формат входных данных

В первой строке записаны от 1 до 20 маленьких английских букв без пробелов.

### Формат выходных данных

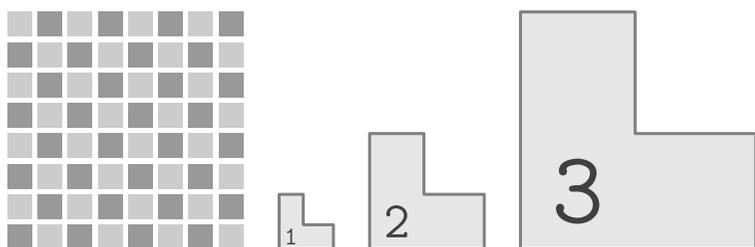
Выясните, какое из слов «yes» и «no» встречается в заданной строке, и выведите это слово. Считается, что слово встретилось в строке, если его буквы идут подряд. Если в строке несколько таких слов, выведите то, которое начинается левее всех. Если же ни одного из этих слов нет, выведите «?».

### Примеры

<i>стандартный ввод</i>	<i>стандартный вывод</i>
abyesson	yes
xnoayesbyesz	no
yneos	?

### Задача С. Уголки

У Миши есть шахматная доска, расчерченная на  $8 \times 8$  клеток, и три картонных уголка. Первый уголок маленький, он закрывает три клетки доски. Второй уголок в два раза больше первого по каждому измерению, он закрывает три квадрата из  $2 \times 2$  клеток. Третий уголок в два раза больше второго по каждому измерению: он закрывает три квадрата, в каждом из которых  $4 \times 4$  клетки. Вот картинка с доской и тремя уголками:



Миша положил уголки на доску так, что ровно одна клетка осталась не закрытой. Затем он сфотографировал доску с уголками: на фотографии видно, какая клетка свободна, но не видно границы уголков. Определите для каждой клетки, какой уголок её закрывает.

#### Формат входных данных

Во входных данных 8 строк, в каждой по 8 символов — описание доски. Символ «#» (решётка) означает, что клетка закрыта уголком. Символ «.» (точка) означает, что клетка не закрыта, такая клетка ровно одна.

#### Формат выходных данных

Выведите 8 строк, в каждой по 8 символов — описание покрытия. Если клетка закрыта уголком, выведите номер этого уголка: 1, 2 или 3. Если клетка не закрыта, выведите 0.

Можно показать, что ответ в этой задаче единственный.

### Пример

стандартный ввод	стандартный вывод	пояснение
##### ##### ##### ##### ##### ##### ##### #####	33331122 33330122 33332222 33332222 33333333 33333333 33333333 33333333	

### Задача D. Капитал

В первый день каникул у Буратино была одна золотая монета. Каждый день Буратино выбирал одно из двух:

- утром посадить все свои монеты на Поле Чудес, а вечером снять с выросших деревьев урожай: в два раза больше монет, чем было утром;
- пойти на весь день в Харчевню Трёх Пескарей, потратив там одну монету.

Известно, что сегодня у Буратино  $n$  золотых монет. Какое минимальное количество дней могло пройти с начала каникул?

#### Формат входных данных

В первой строке записано целое число  $n$  — количество монет у Буратино ( $1 \leq n \leq 10^9$ ).

#### Формат выходных данных

В первой строке выведите одно целое число: минимальное возможное количество дней, прошедших с начала каникул.

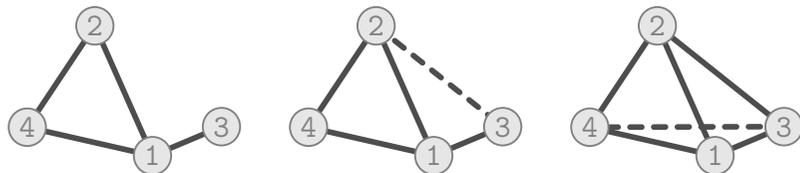
### Примеры

стандартный ввод	стандартный вывод	пояснение
8	3	$1 \xrightarrow{*2} 2 \xrightarrow{*2} 4 \xrightarrow{*2} 8$
5	5	$1 \xrightarrow{*2} 2 \xrightarrow{*2} 4 \xrightarrow{-1} 3 \xrightarrow{*2} 6 \xrightarrow{-1} 5$

## Задача Е. Путь

В Плоской стране  $n$  городов, пронумерованных числами  $1, 2, \dots, n$ . Для каждого города  $i$  известны его координаты на плоскости:  $(x_i, y_i)$ .

Жители Плоской страны решили соединить все города скоростными двусторонними дорогами. Для каждой пары городов  $(i, j)$  они посчитали Евклидово расстояние между ними:  $d(i, j) = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2}$ . Оказалось, что расстояния между всеми парами городов различны.



Далее жители рассматривают пары городов  $(i, j)$  в порядке возрастания  $d(i, j)$ . Для каждой пары они хотят построить дорогу — отрезок прямой между точками  $(x_i, y_i)$  и  $(x_j, y_j)$  (на примере слева показаны первые четыре таких дороги). Если этот отрезок не пересекает никакую построенную дорогу (пунктир на примере в центре), то новая дорога строится, иначе (пунктир на примере справа) — не строится.

Сейчас все возможные дороги уже построены. Для каждой пары городов  $(i, j)$  найдите любой кратчайший путь между ними по дорогам и выведите номер второго (следующего за  $i$ ) города на этом пути.

### Формат входных данных

В первой строке задано целое число  $n$  — количество городов ( $1 \leq n \leq 50$ ).

В следующих  $n$  строках заданы координаты городов. Каждая из этих строк содержит два целых числа  $x_i$  и  $y_i$  ( $1 \leq x_i, y_i \leq 999$ ).

Расстояния между всеми парами городов различны. Никакие два города не находятся в одной точке. Никакие три города не лежат на одной прямой. Каждый тест выбран случайно и равномерно среди всех возможных тестов с заданным  $n$ .

### Формат выходных данных

Выведите  $n$  строк, в каждой по  $n$  чисел. В строке  $i$  число  $j$  — это номер следующего после  $i$  города на каком-то кратчайшем пути из  $i$  в  $j$ . Для  $i = j$  выводите 0.

Пусть для какой-то пары вершин участник предлагает путь длины  $s$ , а кратчайший путь имеет длину  $t$ . Путь участника считается кратчайшим, если  $|s - t|/t < 10^{-6}$ .

## Пример

стандартный ввод	стандартный вывод	пояснение
4 586 635 382 179 830 533 143 535	0 2 3 4 1 0 3 4 1 2 0 1 1 2 1 0	

## Задача Ф. Буквы

Построим бесконечную строку из букв «а» и «б».

Начнём со строки «а». Зафиксируем число  $n$ . Теперь бесконечное количество раз сделаем следующую операцию: возьмём строку, которая у нас есть, и одновременно заменим все буквы «а» на строки «aaa...aab», в которых  $n$  букв «а» и одна буква «б», а все буквы «б» на строки «а». Например, при  $n = 2$  первые несколько операций выглядят так:

«а»  $\rightarrow$  «aab»  $\rightarrow$  «aabaaba»  $\rightarrow$  «aabaabaabaabaabaab»  $\rightarrow$  ...

Выведите подстроку полученной бесконечной строки: от  $\ell$ -й буквы до  $r$ -й включительно. Буквы в строке нумеруются с единицы.

### Формат входных данных

В первой строке заданы три целых числа:  $n$ ,  $\ell$  и  $r$ . Ограничения:

- $1 \leq n \leq 10$
- $1 \leq \ell, r \leq 10^{15}$
- $0 \leq r - \ell \leq 1000$

### Формат выходных данных

В первой строке выведите буквы с  $\ell$ -й до  $r$ -й без пробелов.

### Примеры

стандартный ввод	стандартный вывод
2 1 17	aabaabaabaabaabaab
10 104 111	aaaaaaba