

## Задача А. Не мешай мне перемешивать

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

**Неправильный** способ перемешивать массив длины  $n$  (который часто предлагают новички) звучит так:

Выберем натуральное число  $s$  и будем  $s$  раз выбирать произвольную пару различных индексов в массиве (каждая пара различных индексов выбирается с равной вероятностью) и менять местами элементы в этих позициях.

Убедимся количественно, что этот метод плох. Посчитайте вероятность, что после применения этого метода при данных  $n$  и  $s$  элемент, который изначально находился в  $a$ -й позиции массива в итоге окажется в  $b$ -й позиции.

### Формат входного файла

Во входном файле содержатся 4 целых числа  $n, s, a, b$  ( $1 \leq n \leq 1000$ ;  $1 \leq s \leq 100\,000$ ;  $0 \leq a, b < n$ ) — размер массива, число произвольных обменов, начальный индекс и конечный индекс.

Индексы в массиве нумеруются от 0 до  $n - 1$ .

### Формат выходного файла

Выведите вероятность попадания данного элемента в позицию  $b$ , с точностью не хуже  $10^{-9}$ .

### Примеры

<code>swaps.in</code>	<code>swaps.out</code>
5 1 0 0	0.6
5 1 0 3	0.1

## Задача В. Красное и черное

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

Вы играете в игру по следующим правилам: вы берете колоду из  $r$  красных и  $b$  черных карт, перемешиваете ее, после чего открываете карты по одной. Когда вы открываете красную карту, вы получаете очко, когда открываете черную — теряете. (В процессе игры ваш баланс может стать отрицательным.) В любой момент (в том числе, еще не открыв первую карту) вы можете прекратить игру, и тогда ваш текущий счет станет окончательным. Какого математическое ожидание вашего окончательного счета, если колода было перемешана идеально (все перестановки карт равновероятны), а вы играете оптимально?

### Формат входного файла

Во входном файле содержатся два числа  $r$  и  $b$  ( $0 \leq r, b \leq 5000$ ).

### Формат выходного файла

Выведите математическое ожидание окончательного счета игры при оптимальной стратегии с абсолютной или относительной погрешностью не более  $10^{-9}$ .

### Пример

<code>redisgood.in</code>	<code>redisgood.out</code>
0 7	0.0000000000000000
4 0	4.0000000000000000
5 1	4.166666666666667

## Задача С. Условности

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

Условную вероятность можно продемонстрировать на примере игральных кубиков. Какова условная вероятность, что на двух (стандартных) кубиках выпало 12 при условии, что как минимум на одном из них выпало 6?

Удивительным образом, ответ равен  $\frac{1}{11}$ , поскольку из 36 равновероятных исходов бросания двух кубиков 11 имеют хотя бы одну шестерку, и только 1 из них имеет сумму 12.

Решите обобщенную задачу. Пусть были брошены  $n$  игральных костей, каждая из которых показывает любое число от 1 до  $m$  с равной вероятностью. Какова условная вероятность, что сумма чисел на костях больше или равна  $s$ , при условии, что как минимум на одной кости выпало число  $v$ ?

### Формат входного файла

Во входном файле содержатся числа  $n$ ,  $m$ ,  $v$  и  $s$  ( $1 \leq n, m \leq 50$ ;  $1 \leq v \leq m$ ;  $1 \leq s \leq nm$ ) — количество костей, число граней на каждой из них, известное выпавшее значение и желаемая сумма.

### Формат выходного файла

Выведите искомую условную вероятность с точностью не хуже  $10^{-9}$ .

### Примеры

<code>conditional.in</code>	<code>conditional.out</code>
2 6 6 12	0.09090909090909091
2 6 6 6	1.0

## Задача D. Мышь в лабиринте

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

Задано поле размером  $N \times M$  клеток. Каждая его клетка либо пуста, либо непроходима (содержит стенку), либо содержит сыр.

В одной из пустых клеток поля изначально находится мышь. Каждый ход мышь выбирает из текущей клетки и свободных соседних (по стороне или углу) клеток с равной вероятностью и переходит в нее. Напримр, если мышь стоит не с краю и все соседние клетки свободны, то в вероятность  $\frac{1}{9}$  она останется на месте, или передет в одну из соседних клеток с вероятностью  $\frac{1}{9}$  в каждую.

Если мышь попадает в клетку с сыром, то она съедает его, и процесс заканчивается.

Требуется найти математическое ожидание числа сделанных ходов.

### Формат входного файла

В первой строке записаны размеры поля — числа  $N$  и  $M$ ,  $1 \leq N, M \leq 20$ .

Далее идут  $N$  строк по  $M$  символов, содержащие описание поля. Символ '.' обозначает пустую клетку, '#' — стенку, 'C' — сыр, 'M' — клетку с мышью. Гарантируется, что будет ровно одна клетка с символом 'M'.

### Формат выходного файла

Выведите искомое математическое ожидание числа ходов или «-1», если съесть сыр невозможно. Точность должна быть не хуже  $10^{-4}$ .

### Примеры

<code>mouse.in</code>	<code>mouse.out</code>
1 2 MC	2.000000000
1 3 M.C	7.000000000
3 3 ### MC# #. #	3.000000000
2 3 M#C .#.	-1
5 5 CCCCC ...# ..M.# .#### #C.#.	10.819526627