



# Общая информация по задачам В, С, D и E

## Ограничение по памяти

Во всех задачах ограничение составляет 256 МБ.

## Ограничение на размер исходного кода программы

Во всех задачах размер файла с исходным кодом решения не должен превышать 256 КБ.

## Ограничение на посылку решений

По каждой задаче на проверку принимается не более 50 решений.

По каждой задаче участник не может отправить решение более одного раза в течение 30 секунд. Это ограничение не распространяется на последние 15 минут соревнований.

## Система оценки

Каждая задача олимпиады поделена на несколько подзадач. Чтобы набрать баллы по подзадаче, программа должна пройти все тесты этой подзадачи.

За каждую задачу выставляется суммарный балл по всем ее подзадачам. В каждой подзадаче оценивается лучшее решение, то есть за подзадачу выставляется максимальный набранный по ней балл среди всех решений.

## Получение информации о результатах проверки

Чтобы получить информацию о проверке вашего решения, используйте ссылку «Информация о проверке» во вкладке «Решения» в PCMS2 Web Client. По каждой задаче вам будет доступна информация по количеству набранных баллов в каждой подзадаче или результат проверки на первом непройденном тесте.

## Таблица результатов

Во время соревнования доступна текущая таблица результатов. Для доступа к ней используйте ссылку «Результаты» в PCMS2 Web Client. Таблица результатов в PCMS2 Web Client не является окончательной.



## Задача А. Пеш-хматы

Ограничение по времени: 1 секунда

Сергею подарили игру «Пеш-хматы». В этой игре предоставлено шахматное поле размером  $w \times h$ . На нем расставлены черные пешки и одна белая.

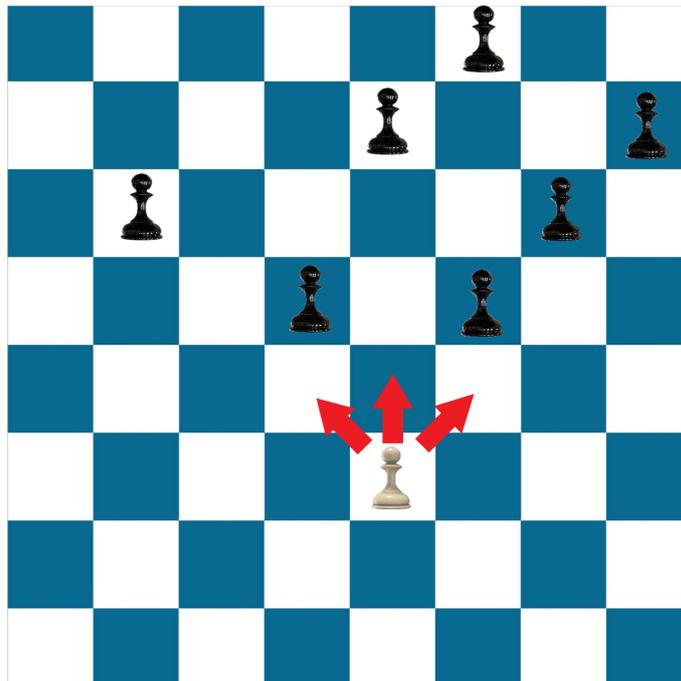
Черные пешки не могут двигаться, в то время как белая может ходить прямо вверх ( $U$ ), по диагонали вправо и вверх ( $R$ ) и по диагонали влево и вверх ( $L$ ). В этой игре пешка может взять любую черную фигуру, которая расположена впереди неё на одну клетку прямо вверх или по диагонали.

Задача игрока состоит в том, чтобы составить маршрут для белой пешки так, чтобы она съела как можно больше черных фигур.

Маршрут описывается последовательностью букв:

- $U$  — пешка идет вперед,
- $R$  — пешка идет по диагонали вправо,
- $L$  — пешка идет по диагонали влево.

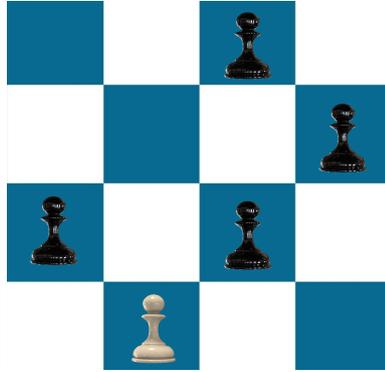
Рис. 1: Например, для данного поля ответом может быть  $URRR$ .



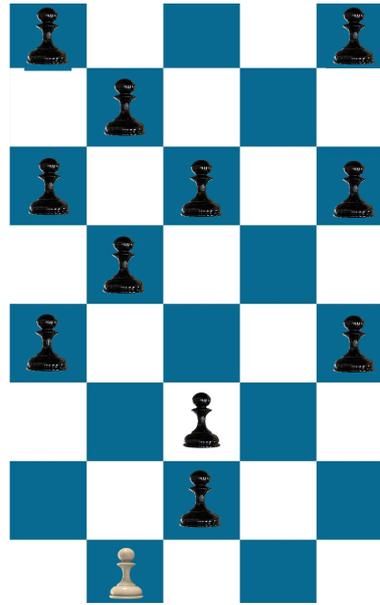
Даны четыре тестовых картинки, для которых требуется найти маршрут. Если вы не знаете ответ для теста, запишите  $X$ . Как ответ на эту задачу прикрепите код или `txt` файл. Если отправляете `txt` файл то на первой строчке будет записан маршрут для первого теста, во второй строчке будет записан маршрут для второго теста и тд. Если отправляете код то на первой строке выходных данных выведете маршрут для первого теста, во второй строчке будет записан маршрут для второго теста и тд.

По этой задаче на проверку принимается не более трех файлов.

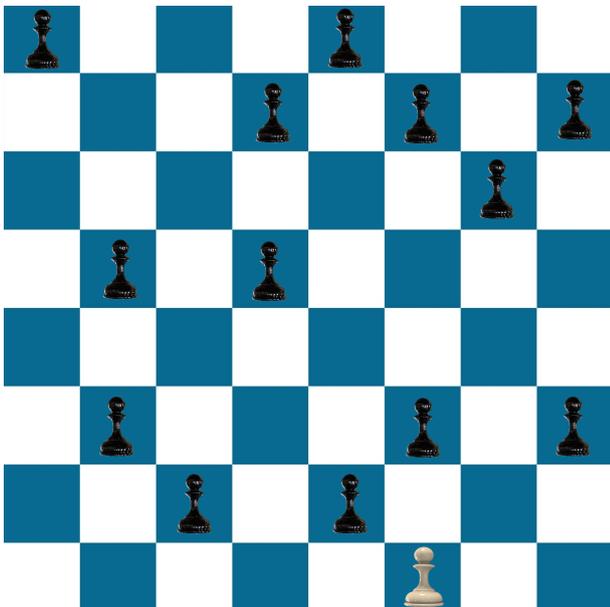
Каждый тест оценивается от 0 до 25 баллов по формуле  $25 \cdot \frac{points_{participant}}{points_{max}}$ , где  $points_{participant}$  — количество съеденных фигур маршрутом участника,  $points_{max}$  — количество съеденных фигур маршрутом жюри. Если маршрут содержит символы кроме  $L, R, U, X$  или выходит за границы поля, то оценивается он будет 0 баллов.



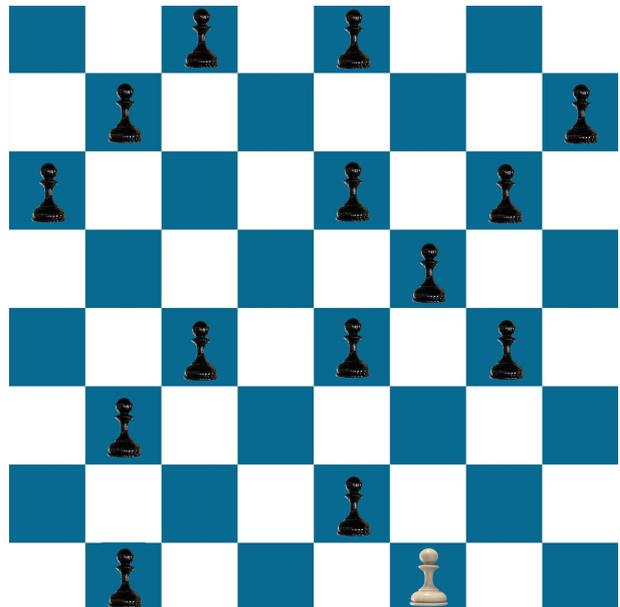
(a) Тест 1



(b) Тест 2



(c) Тест 3



(d) Тест 4

Номер теста	Ответ
1	(поле ответа)
2	(поле ответа)
3	(поле ответа)
4	(поле ответа)



## Задача В. Цирк

Ограничение по времени: 2 секунды

В честь дня рождения цирка клоун Пилюлькин раздает детям бесплатные билеты на свободные места, которые остались после продажи билетов. Арена цирка разделена на четыре сектора  $A$ ,  $B$ ,  $C$  и  $D$ .

Пилюлькин раздает билеты по следующему принципу: первому ребенку достается билет в сектор  $A$ , второму — в сектор  $B$ , третьему — в сектор  $C$ , четвертому — в сектор  $D$ , пятому опять в сектор  $C$ , шестому — в сектор  $B$  и так далее (Последовательность секторов имеет вид:  $A - B - C - D - C - B - A - B - C - D - \dots$ ).

Как только у Пилюлькина не будет возможности подарить билет в очередной сектор, он прекращает бесплатную раздачу билетов и начинает представление.

Сколько детей смогут посетить цирк бесплатно, если никаким другим способом бесплатный билет получить нельзя?

### Формат входных данных

В первой строке входного файла задано целое число  $A$  ( $0 \leq A \leq 2 \cdot 10^{16}$ ) — количество билетов в сектор  $A$ .

Во второй строке входного файла задано целое число  $B$  ( $0 \leq B \leq 2 \cdot 10^{16}$ ) — количество билетов в сектор  $B$ .

В третьей строке входного файла задано целое число  $C$  ( $0 \leq C \leq 2 \cdot 10^{16}$ ) — количество билетов в сектор  $C$ .

В четвертой строке входного файла задано целое число  $D$  ( $0 \leq D \leq 2 \cdot 10^{16}$ ) — количество билетов в сектор  $D$ .

### Формат выходных данных

Выведите количество билетов, которые сможет раздать Пилюлькин до начала представления.

### Система оценки

Баллы за каждую подзадачу начисляются только в случае, если все тесты для этой подзадачи и необходимых подзадач успешно пройдены.

Подзадача	Баллы	Ограничения	Необходимые подзадачи	Информация о проверке
0	0	Тесты из условия	—	полная
1	15	$A = B = C = D \leq 10^6$	—	первая ошибка
2	25	$A, B, C, D \leq 10^6$	1	первая ошибка
3	15	$A = B = C = D \leq 2 \cdot 10^{16}$	1	первая ошибка
4	45	Нет дополнительных ограничений	1 — 3	первая ошибка



## Примеры

стандартный ввод	стандартный вывод
1 1 1 1	4
1 2 3 4	6
1 2 2 4	6
23 10 3 99	10



## Задача С. Слова

Ограничение по времени: 2 секунды

Вы упражняетесь в слепой печати, и сегодня у вас по плану напечатать  $m$  слов. Для этого вы можете выбрать произвольные  $m$  слов из набора  $s_1, \dots, s_n$  ( $1 \leq n \leq 2 \cdot 10^5$ ), и напечатать их через пробел. Причем каждое слово может быть выбрано не более одного раза.

Кроме того, в каждом из слов набора могут встречаться только первые  $k$  ( $1 \leq k \leq 26$ ) букв латинского алфавита.

Вы знаете, сколько времени у вас занимает перенести палец с одной буквы на клавиатуре на другую. Более формально, известна матрица  $t_{i,j}$  ( $1 \leq i, j \leq k$ ), где  $t_{i,j}$  равняется времени, которое вы потратите на перенос пальца с  $i$ -го символа на  $j$ -й символ. В этой задаче мы пренебрежем временем на перенос пальца с символа на пробел и наоборот.

Таким образом, чтобы напечатать два слова «ab» и «b» потребуется перенести палец с буквы «a» на букву «b», потратив  $t_{1,2}$  времени, после чего мгновенно перенести палец с буквы «b» на пробел и затем мгновенно перенести палец с пробела на букву «b». Значит, суммарно потребуется  $t_{1,2}$  времени.

Вы хотите напечатать  $m$  слов из набора таким образом, чтобы минимизировать затраченное время. Найдите это время.

### Формат входных данных

В первой строке записано натуральное число  $n$  ( $1 \leq n \leq 2 \cdot 10^5$ ) — количество слов в наборе.

Во второй строке записано  $n$  строк  $s_1, \dots, s_n$ , каждая из которых состоит из строчных латинских букв.

В третьей строке записано натуральное число  $m$  ( $1 \leq m \leq n$ ) — количество слов, которые необходимо выбрать для печати.

В четвертой строке записано натуральное число  $k$  ( $1 \leq k \leq 26$ ), задающее множество первых  $k$  латинских букв, из которых могут состоять слова.

В следующих строках записана матрица  $t_{i,j}$  ( $1 \leq t_{i,j} \leq 10^9$ ) размера  $k \times k$ , состоящая из неотрицательных целых чисел.

Гарантируется, что суммарная длина всех слов не превосходит  $2 \cdot 10^5$ .

### Формат выходных данных

Выведите единственное число — минимальное время, необходимое для набора  $m$  слов.

### Система оценки

Баллы за каждую подзадачу начисляются только в случае, если все тесты для этой подзадачи и необходимых подзадач успешно пройдены.

Подзадача	Баллы	Ограничения	Необходимые подзадачи	Информация о проверке
0	0	Тесты из условия	—	полная
1	20	$k = 1$	—	первая ошибка
2	20	Сумма длин всех слов не больше 1000	—	первая ошибка
3	30	$n \leq 1000$	2	первая ошибка
4	30	Нет дополнительных ограничений	1 — 3	первая ошибка



## Примеры

стандартный ввод	стандартный вывод
2 ab ba 1 2 0 1 2 0	1
3 a aa aaa 2 1 5	5



## Задача D. Пеш-хматы

Ограничение по времени: 2 секунды

Сергею подарили игру «Пеш-хматы». В этой игре предоставлено шахматное поле размером  $w \times h$ . Клетка  $(1; 1)$  расположена в левом нижнем углу. На нем расставлены черные пешки.

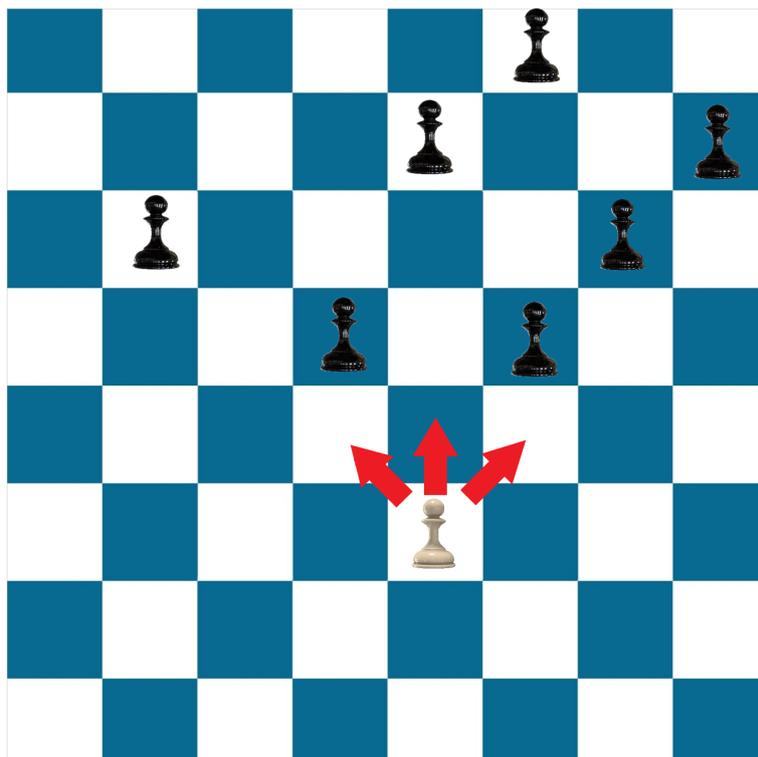
Черные пешки не могут двигаться, в то время как белая может ходить прямо вверх ( $U$ ), по диагонали вправо и вверх ( $R$ ) и по диагонали влево и вверх ( $L$ ). В этой игре пешка может взять любую черную фигуру, которая расположена впереди неё на одну клетку прямо вверх или по диагонали.

Задача игрока состоит в том, чтобы выбрать куда поставить пешку и составить маршрут для белой пешки так, чтобы она съела как можно больше черных фигур. Белую пешку нельзя ставить на поле, где есть черная пешка.

Маршрут описывается последовательностью букв:

- $U$  — пешка идет вперед,
- $R$  — пешка идет по диагонали вправо,
- $L$  — пешка идет по диагонали влево.

Рис. 2: Например, для данного поля ответом будет  $URRR$ .



### Формат входных данных

В первой строке входного файла задано два целых числа  $h$  и  $w$  ( $1 \leq w, h \leq 2 \cdot 10^5$ ) — размеры шахматного поля.

Гарантируется, что  $w \cdot h \leq 2 \cdot 10^5$  и что существует свободная/пустая клетка.

В каждой из следующих  $h$  строк записано по  $w$  символов  $a_{ij}$  ( $a_{ij} \in \{*, B\}$ ), где

- $*$  — пустая клетка,
- $B$  — на клетке стоит черная пешка.



## Формат выходных данных

Выведите в первой строке два целых числа, куда поставить пешку (номер строки и номер столбца).

Во второй строке выведите количество съеденных пешек.

Если пешка съела хотя бы одну фигуру, то на третьей строке выведите маршрут, чтобы белая пешка съела наибольшее количество черных фигур.

Если ответов несколько, то выведите любой.

## Система оценки

Баллы за каждую подзадачу начисляются только в случае, если все тесты для этой подзадачи и необходимых подзадач успешно пройдены.

Подзадача	Баллы	Ограничения	Необходимые подзадачи	Информация о проверке
0	0	Тесты из условия	—	полная
1	30	$h, w \leq 10$	—	первая ошибка
2	35	$h, w \leq 30$	1	первая ошибка
3	35	Нет дополнительных ограничений	1 — 2	первая ошибка

## Пример

стандартный ввод	стандартный вывод
<pre>8 8 *****B** ****B**B *B****B* ***B*B** ***** ***** ***** *****</pre>	<pre>4 3 3 RURR</pre>



## Задача Е. Плейлист

Ограничение по времени: 2 секунды

Единорог Спаркс очень любит прослушивать свой плейлист в социальной сети ВКонтакте. Для разнообразия Спаркс каждый раз выбирает новый порядок песен, и этот раз не стал исключением. Спаркс решил выбрать такой порядок песен, чтобы первое время продолжительность песен не убывала, а потом не возрастала. Помогите Спарксу — найдите количество способов перемешать песни так, чтобы Спарксу было интересно слушать плейлист.

Более формально: дан массив  $a_1, \dots, a_n$  ( $1 \leq a_i \leq 10^9$ ), где  $a_i$  — это продолжительность  $i$ -й песни. Требуется найти количество способов переупорядочить элементы массива  $a$  таким образом, чтобы существовал такой индекс  $k$ , что  $a_i \leq a_{i+1}$  для всех  $1 \leq i \leq k-1$ , а также  $a_i \geq a_{i+1}$  для всех  $k \leq i \leq n-1$ .

Так как ответ может быть очень большим, выведите его остаток от деления на  $10^9 + 7$ .

### Формат входных данных

В первой строке входного файла задано число  $n$  ( $1 \leq n \leq 2 \cdot 10^5$ ) — количество песен в плейлисте Спаркса.

Во второй строке входного файла задано  $n$  чисел  $a_1, \dots, a_n$  ( $1 \leq a_i \leq 10^9$ ), где  $a_i$  — это продолжительность  $i$ -й песни.

### Формат выходных данных

Выведите единственное число — ответ на задачу по модулю  $10^9 + 7$ .

### Система оценки

Баллы за каждую подзадачу начисляются только в случае, если все тесты для этой подзадачи и необходимых подзадач успешно пройдены.

Подзадача	Баллы	Ограничения	Необходимые подзадачи	Информация о проверке
0	0	Тесты из условия	—	полная
1	10	$n \leq 10$	—	первая ошибка
2	20	Все $a_i$ попарно различны	—	первая ошибка
3	20	$n \leq 20$	1	первая ошибка
4	50	Нет дополнительных ограничений	1 — 3	первая ошибка

### Примеры

стандартный ввод	стандартный вывод
3 1 2 3	4
3 2 1 1	6