

## Задача А. Индусская математика

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

Если вы поступаете на продвинутый курс C# или на Шпору, решать все задачи нужно на C#. Если вы поступаете на базовый курс C#, можно использовать любой язык из доступных.

Пусть задано число  $x$ , десятичная запись которого не содержит нулей. Рассмотрим все числа, которые можно получить из числа  $x$  перестановкой его цифр. Пусть  $M$  — наибольшее из таких чисел, а  $m$  — наименьшее. Определим число  $K(x)$  как  $M - m$ .

Например, для  $x = 6174$  получаем  $M = 7641$ ,  $m = 1467$ ,  $K(x) = 7641 - 1467 = 6174$ .

Вычислите  $K(x)$  по заданному числу  $x$ .

### Формат входных данных

В единственной строке записано целое число  $x$  ( $1 \leq x \leq 10^9 - 1$ ). Десятичная запись  $x$  не содержит нулей.

### Формат выходных данных

Выведите целое число  $K(x)$ .

### Система оценки

В этой задаче одна группа тестов стоимостью 10 баллов.

### Пример

стандартный ввод	стандартный вывод
6174	6174

## Задача В. Недостающее число

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

Если вы поступаете на продвинутый курс C# или на Шпору, решать все задачи нужно на C#. Если вы поступаете на базовый курс C#, можно использовать любой язык из доступных.

Дан массив из нечётного количества целых чисел. Массив обладает следующим свойством — если дополнить его ещё одним числом, все числа в массиве можно будет разбить на пары так, что сумма в каждой паре будет равна нулю.

Скажите, какого числа не хватает в массиве.

### Формат входных данных

В первой строке записано нечётное целое число  $n$  ( $1 \leq n \leq 99\,999$ ). Далее в  $n$  строках записаны целые числа  $a_1, \dots, a_n$  — исходный массив ( $1 \leq |a_i| \leq 1000$ ).

### Формат выходных данных

Выведите целое число, которое нужно добавить в массив, чтобы все числа массива разбивались на пары с нулевой суммой. Гарантируется, что такое число существует.

### Система оценки

В этой задаче две группы тестов.

Первая группа тестов стоит 6 баллов, для неё выполняется ограничение  $n \leq 999$ .

Вторая группа тестов стоит 4 балла, для неё выполняется ограничение  $n \leq 99\,999$ .

### Пример

стандартный ввод	стандартный вывод
7 1 -2 1 2 3 -1 -1	-3

## Задача С. Премьер-лига

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

Если вы поступаете на продвинутый курс C# или на Шпору, решать все задачи нужно на C#. Если вы поступаете на базовый курс C#, можно использовать любой язык из доступных.

Завершается чемпионат Российской футбольной премьер-лиги. Сыграны все матчи, кроме последнего. В последнем матче команда А принимает у себя дома команду В. От результата этого матча может зависеть, какое место займёт команда А по итогам чемпионата.

Определите, на каком месте закончит чемпионат команда А в случае победы, ничьей или поражения в своём последнем матче.

По правилам футбола, если матч заканчивается победой одной из команд, то победившая команда получает три очка, а проигравшая — ноль очков. Если матч заканчивается ничьей, то обе команды получают по одному очку. Команды ранжируются в турнирной таблице в порядке убывания набранных очков, в случае равенства очков у двух и более команд они ранжируются по алфавиту.

### Формат входных данных

В первой строке записано целое число  $n$  — количество команд в премьер-лиге ( $2 \leq n \leq 20$ ). В следующих  $n$  строках приводится положение команд в турнирной таблице до того, как состоялся последний матч, в формате «название команды» «количество набранных очков». Названия команд имеют длину от 3 до 12 и состоят только из заглавных латинских букв. Гарантируется, что команды в таблице упорядочены по убыванию количества очков, а при равенстве очков — по алфавиту. Названия всех команд различны. Количество очков каждой команды лежит в пределах от 0 до 99.

Распределение очков между командами не обязательно описывает ситуацию, возможную в реальности — в рамках данной задачи не нужно обращать внимание на это и проверять входные данные на корректность.

В последней строке входных данных написано, какой матч ещё не сыгран. Его описание имеет формат «название команды А»-«название команды В». Названия команд А и В различны и присутствуют в турнирной таблице.

### Формат выходных данных

Выведите три целых числа через пробел — на каком месте закончит чемпионат команда А в случае победы, ничьей или поражения в матче с командой В.

### Система оценки

В этой задаче две группы тестов.

Первая группа тестов стоит 5 баллов, для тестов в ней разница между командами А и В в турнирной таблице **превышает три очка**.

Вторая группа тестов стоит 5 баллов, для тестов в ней разница между командами А и В в турнирной таблице может быть произвольной.

## Примеры

стандартный ввод	стандартный вывод
7 ZENIT 65 SOCHI 56 DINAMO 53 CSKA 50 KRASNODAR 49 LOKOMOTIV 48 AKHMAT 41 KRASNODAR-AKHMAT	4 5 5
2 ARGENTINA 5 JAMAICA 0 ARGENTINA-JAMAICA	1 1 1

## Задача D. Детская площадка

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

Если вы поступаете на продвинутый курс C# или на ШПору, решать все задачи нужно на C#. Если вы поступаете на базовый курс C#, можно использовать любой язык из доступных.

Дана схема городского квартала в виде прямоугольной таблицы размером  $n \times m$ . Каждая ячейка таблицы или принадлежит какому-то зданию, или свободна. Все ячейки по границам квартала заняты зданиями.

Нужно разместить в этом квартале детскую площадку в виде прямоугольника, стороны которого параллельны границам квартала. Все ячейки, которые будут отданы под площадку, должны быть свободными. Если есть несколько вариантов размещения площадки, нужно выбрать такой, при котором она имеет наибольшую площадь (то есть занимает наибольшее число ячеек таблицы на схеме).

### Формат входных данных

В первой строке записаны целые числа  $n$  и  $m$  — размеры таблицы ( $3 \leq n, m \leq 500$ ). В следующих  $n$  строках по  $m$  символов в каждой записана таблица. Свободные ячейки обозначены в ней символами «.», а занятые — символами «\*». Гарантируется, что первая и последняя строка таблицы, а также первый и последний столбец таблицы заполнены символами «\*».

### Формат выходных данных

Выведите наибольшую возможную площадь (количество ячеек) детской площадки. Если в квартале невозможно разместить детскую площадку, выведите «0».

### Система оценки

В этой задаче три группы тестов.

Первая группа тестов стоит 4 балла, для неё выполняются ограничения  $n, m \leq 30$ .

Вторая группа тестов стоит 4 балла, для неё выполняются ограничения  $n, m \leq 100$ .

Третья группа тестов стоит 2 балла, для неё выполняются ограничения  $n, m \leq 500$ .

### Примеры

стандартный ввод	стандартный вывод
4 10 ***** *...*...* *...*...* *****	8
3 3 *** *** ***	0

## Задача Е. Агитация

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

Если вы поступаете на продвинутый курс C# или на Шпору, решать все задачи нужно на C#. Если вы поступаете на базовый курс C#, можно использовать любой язык из доступных.

В уездном городе Е. скоро пройдут выборы. Штаб одного из кандидатов напечатал плакаты, агитирующие за своего кандидата. В штабе работает студент Архип, ему-то и поручили разместить плакаты на улицах города.

В первый день Архип ходил по городу и наклеивал плакаты на стены тех зданий, которые ему понравились. Но руководитель штаба остался недоволен работой Архипа. Поэтому утром второго дня руководитель написал на каждом плакате, на какой улице его нужно разместить, и сказал Архипу расклеивать плакаты ровно в том порядке, в котором они лежат в стопке.

Архип решил действовать по следующему алгоритму:

1. Если стопка не пуста, взять из неё очередной плакат.
2. Прочитать на плакате название улицы, на которой его нужно разместить.
3. Пойти на указанную улицу и обходить все здания на ней по возрастанию номеров, начиная с номера 1, до тех пор, пока он не дойдёт до здания, на котором ещё нет плаката (т.е. плакат не был наклеен ни в первый, ни во второй день).
4. Наклеить плакат на стену этого здания и вернуться к шагу 1.

Определите для каждого плаката из стопки номер здания на улице, на котором он будет размещён. Можете считать, что на каждой улице города Е. достаточное количество зданий, и все они занумерованы последовательными натуральными числами, начиная с единицы, без пропусков и дублей.

### Формат входных данных

В первой строке записано целое число  $n$  — количество зданий, на которых Архип разместил плакаты в первый день ( $0 \leq n \leq 90\,000$ ). Далее в  $n$  строках перечислены адреса этих зданий. Каждый адрес — это название улицы и номер дома, которые склеены в одну строку без пробелов и прочих разделителей. Название улицы состоит из строчных латинских букв и имеет длину от 1 до 1 500. Номер дома — это целое число от 1 до  $10^6$  без ведущих нулей. Все адреса попарно различны.

В следующей строке записано целое число  $m$  — количество плакатов в стопке на второй день ( $1 \leq m \leq 300\,000$ ). Далее в  $m$  строках перечислены названия улиц, на которых следует разместить эти плакаты (именно в таком порядке). Названия улиц состоят из строчных латинских букв и имеют длину от 1 до 1 500.

Суммарный размер входных данных не превышает одного мегабайта.

### Формат выходных данных

Для каждого плаката в стопке определите номер здания на соответствующей улице, на котором он будет размещён.

### Система оценки

В этой задаче две группы тестов.

Первая группа тестов стоит 5 баллов, для неё выполняется ограничение  $n \leq 5$ .

Вторая группа тестов стоит 5 баллов, для неё выполняется ограничение  $n \leq 90\,000$ .

## Пример

стандартный ввод	стандартный вывод
3	2
mira32	1
turgeneva4	2
miral	3
6	5
mira	1
turgeneva	
turgeneva	
turgeneva	
turgeneva	
lenina	