

To print higher-resolution math symbols, click the
Hi-Res Fonts for Printing button on the jsMath control panel.

Условия задач

Сайт: [Дистанционная подготовка](#)
Курс: Структуры данных
Условия задач: Условия задач
Printed by: Гость
Date: Понедельник 19 Март 2018, 19:51

Список задач

- [Задача А. Списки по классам](#)
 - [Задача В. Игра в пьяницу](#)
 - [Задача С. Правильная скобочная последовательность](#)
 - [Задача D. Постфиксная запись](#)
 - [Задача Е. Гоблины и шаманы](#)
 - [Задача F. Контейнеры](#)
-

Списки по классам

Задача А. Списки по классам

Формат входных данных

В каждой строке сначала записан номер класса (число, равное 9, 10 или 11), затем (через пробел) – фамилия ученика. Общее число строк в файле не превосходит 100000. Длина каждой фамилии не превосходит 50 символов.

Формат выходных данных

Необходимо вывести список школьников по классам: сначала всех учеников 9 класса, затем – 10, затем – 11. Внутри одного класса порядок вывода фамилий должен быть таким же, как на входе.

Пример

Входные данные	Выходные данные
9 Иванов 10 Петров 11 Сидоров 9 Григорьев 9 Сергеев 10 Яковлев	9 Иванов 9 Григорьев 9 Сергеев 10 Петров 10 Яковлев 11 Сидоров

Тесты - в кодировке KOI-8

Игра в пьяницу

Задача В. Игра в пьяницу

В игре в пьяницу карточная колода раздается поровну двум игрокам. Далее они вскрывают по одной верхней карте, и тот, чья карта старше, забирает себе обе вскрытые карты, которые кладутся под низ его колоды. Тот, кто остается без карт – проигрывает.

Для простоты будем считать, что все карты различны по номиналу, а также, что самая младшая карта побеждает самую старшую карту ("шестерка берет туза").

Игрок, который забирает себе карты, сначала кладет под низ своей колоды карту первого игрока, затем карту второго игрока (то есть карта второго игрока оказывается внизу колоды).

Напишите программу, которая моделирует игру в пьяницу и определяет, кто выигрывает. В игре участвует 10 карт, имеющих значения от 0 до 9, большая карта побеждает меньшую, карта со значением 0 побеждает карту 9.

Входные данные

Программа получает на вход две строки: первая строка содержит 5 чисел, разделенных пробелами — номера карт первого игрока, вторая – аналогично 5 карт второго игрока. Карты перечислены сверху вниз, то есть каждая строка начинается с той карты, которая будет открыта первой.

Выходные данные

Программа должна определить, кто выигрывает при данной раздаче, и вывести слово first или second, после чего вывести количество ходов, сделанных до выигрыша. Если на протяжении 10^6 ходов игра не заканчивается, программа должна вывести слово botva.

Примеры

входные данные
1 3 5 7 9 2 4 6 8 0
выходные данные
second 5

Правильная скобочная последовательность

Задача С. Правильная скобочная последовательность

Рассмотрим последовательность, состоящую из круглых, квадратных и фигурных скобок. Программа должна определить, является ли данная скобочная последовательность правильной.

Пустая последовательность является правильной. Если A – правильная, то последовательности (A) , $[A]$, $\{A\}$ – правильные. Если A и B – правильные последовательности, то последовательность AB – правильная.

Входные данные

В единственной строке записана скобочная последовательность, содержащая не более 100000 скобок.

Выходные данные

Если данная последовательность правильная, то программа должна вывести строку `yes`, иначе строку `no`.

Примеры

входные данные
<code>()[]</code>
выходные данные
<code>yes</code>

Постфиксная запись

Задача D. Постфиксная запись

В постфиксной записи (или обратной польской записи) операция записывается после двух операндов. Например, сумма двух чисел A и B записывается как $A B +$. Запись $B C + D *$ обозначает привычное нам $(B + C) * D$, а запись $A B C + D * +$ означает $A + (B + C) * D$. Достоинство постфиксной записи в том, что она не требует скобок и дополнительных соглашений о приоритете операторов для своего чтения.

Входные данные

В единственной строке записано выражение в постфиксной записи, содержащее однозначные числа и операции $+$, $-$, $*$.

Выходные данные

Необходимо вывести значение записанного выражения.

Примеры

входные данные
8 9 + 1 7 - *
выходные данные
-102

Гоблины и шаманы

Задача Е. Гоблины и шаманы

Гоблины Мглистых гор очень любят ходить к своим шаманам. Так как гоблинов много, к шаманам часто образуются очень длинные очереди. А поскольку много гоблинов в одном месте быстро образуют шумную толпу, которая мешает шаманам проводить сложные медицинские манипуляции, последние решили установить некоторые правила касательно порядка в очереди.

Обычные гоблины при посещении шаманов должны вставать в конец очереди. Привилегированные же гоблины, знающие особый пароль, встают ровно в ее середину, причем при нечетной длине очереди они встают сразу за центром.

Так как гоблины также широко известны своим непочтительным отношением ко всяческим правилам и законам, шаманы попросили вас написать программу, которая бы отслеживала порядок гоблинов в очереди.

Входные данные

В первой строке входных данных записано число N ($1 \leq N \leq 10^5$) - количество запросов к программе. Следующие N строк содержат описание запросов в формате:

- "+ i" - гoblin с номером i ($1 \leq i \leq N$) встает в конец очереди.
- "* i" - привилегированный гoblin с номером i встает в середину очереди.
- "-" - первый гoblin из очереди уходит к шаманам. Гарантируется, что на момент такого запроса очередь не пуста.

Выходные данные

Для каждого запроса типа "-" программа должна вывести номер гоблина, который должен зайти к шаманам.

Примеры

входные данные
7 + 1 + 2 - + 3 + 4 - -
выходные данные
1 2 3

Контейнеры

Задача F. Контейнеры

На складе хранятся контейнеры с товарами N различных видов. Все контейнеры составлены в N стопок. В каждой стопке могут находиться контейнеры с товарами любых видов (стопка может быть изначально пустой).

Автопогрузчик может взять верхний контейнер из любой стопки и поставить его сверху в любую стопку. Необходимо расставить все контейнеры с товаром первого вида в первую стопку, второго вида – во вторую стопку и т.д.

Программа должна вывести последовательность действий автопогрузчика или сообщение о том, что задача решения не имеет.

Формат входных данных

В первой строке входных данных записано одно натуральное число N , не превосходящее 500. В следующих N строках описаны стопки контейнеров: сначала записано число k_i – количество контейнеров в стопке, а затем k_i чисел – виды товара в контейнерах в данной стопке, снизу вверх. В каждой стопке вначале не более 500 контейнеров (в процессе переноса контейнеров это ограничение может быть нарушено).

Формат выходных данных

Программа должна вывести описание действий автопогрузчика: для каждого действия напечатать два числа – из какой стопки брать контейнер и в какую стопку класть. (Обратите внимание, что минимизировать количество операций автопогрузчика не требуется.) Если задача не имеет решения, необходимо вывести одно число 0. Если контейнеры изначально правильно размещены по стопкам, то выводить ничего не нужно.

Пример

Входные данные	Выходные данные
3 4 1 2 3 2 0 0	1 2 1 3 1 2

Объяснение примера. Изначально в первой стопке лежат четыре контейнера – снизу контейнер с товаром первого вида, над ним – с товаром второго вида, над ним третьего, и сверху еще один контейнер с товаром второго вида. Вторая и третья стопки – пусты.