

Задания финального тура

Олимпиада по веб-программированию, Академия «1С-Битрикс», фирма «1С», 2020 год

А. Заказы

Имя входного файла: input.txt или стандартный поток ввода
Имя выходного файла: output.txt или стандартный поток вывода

Владелец интернет-магазина, программист, филантроп Евгений подключил свой магазин к торговым площадкам Яндекс.Маркет, Беру, Авито и Юле. Выгрузив товары он получил небывалое количество заказов. Всё бы хорошо, да только товаров на все заказы не хватит. Поэтому Евгений решил отправлять заказы в порядке их поступления и только те, которые он мог полностью обеспечить.

Входные данные

На вход подаются остатки на складах в виде "<id_товара> <кол-во>" по одному товару в строке.

Далее через одну строку идёт список заявок, поступивших в учетную систему в виде "<id_заявки> <id_товара_1> <кол-во_1> <id_товара_2> <кол-во_2> ..." по одной заявке в строке.

Заявки отсортированы в порядке поступления в учетную систему. Если заявка может быть обработана, товар резервируется и соответствующее количество вычитается из остатков. Если в заявке есть хотя бы один товар, отсутствующий на складе - заявка отклоняется полностью и изменение остатков не изменяются

Выходные данные

Требуется вывести остатки по всем товарам после обработки всех заявок. Выводить в том же порядке и в том же формате, что и на входе.

Пример

Входные данные	Результат работы
1 10 2 10 3 10 1 1 3 2 7 3 1 2 1 8 2 1 3 4 6	1 7 2 3 3 9

В. Комментарии

Имя входного файла: input.txt или стандартный поток ввода
Имя выходного файла: output.txt или стандартный поток вывода

Администратор городского портала Борис разработал систему управления древовидными комментариями на сайте. В его обязанности входит не только поддержка системы в рабочем состоянии, но и модерация сообщений.

Система позволяет оставлять комментарии верхнего уровня, ответы на другие комментарии. А также администратор может заморозить или разморозить определенную подветку комментариев. В замороженную ветку нельзя добавлять новые комментарии.

На вход в каждой строке подаются по одной из возможных команд. Каждая команда описывается набором полей через пробел, где первое поле - название команды.

Возможные команды:

1. add id parent_id - добавить комментарий с идентификатором id в ветку комментария parent_id. У комментариев верхнего уровня в поле parent_id будет символ '-'.
2. freeze id - заморозить подветку комментария с идентификатором id. Если передан id '-' - заморозить все комментарии.
3. unfreeze id - разморозить подветку комментария с идентификатором id. Если передан id '-' - разморозить все комментарии.

Гарантируется, что нет двух команд add с одним и тем же id.

Входные данные

По одному комментарию в строке, комментарий определяется своим id.

Выходные данные

Вывести дерево комментариев, получившееся после выполнения всех команд.

На одном уровне комментарии сортируются в порядке добавления.

Для каждого комментария следом за ним вывести все комментарии из его подветки.

Перед комментарием вывести столько прочерков, на каком уровне вложенности он находится, после комментария вывести звездочку (*) если он находится в замороженной ветке.

Пример

Входные данные	Результат работы
----------------	------------------

add 1 -	1
add 2 -	-3
freeze 2	--4
add 3 1	2
add 4 3	-6*
add 5 2	
unfreeze -	
add 6 2	
freeze 6	

С. Стили

Имя входного файла: input.txt или стандартный поток ввода
Имя выходного файла: output.txt или стандартный поток вывода

У интернет-магазина тормозных колодок и дисков появилась потребность в ускорение ряда страниц сайта. Разработчики решили начать с оптимизации CSS. Поскольку страницы легковесные, браузер тратить много времени на установку соединения с сервером чтобы скачать CSS файлы. Разработчики решили сделать Инлайн-CSS.

Даны CSS стили и HTML страница, необходимо заинлайнить стили и вернуть получившуюся HTML страницу.

Надо выбрать стили с простыми селекторами по одному классу или по одному id (#id {*правила*} или .class {*правила*}), и заинлайнить только те из них, которые применяются не менее чем к N элементам на странице. Заинлайнить можно не более M правил в один элемент. Если общее количество правил, которые надо заинлайнить в элемент, больше M - не инлайнятся никакие. Анализировать сами правила не надо.

Входные данные

В первой строке целые числа N и M через пробел.

В следующей строке указывается "<<<CSS>>>" — признак начала css-правил.

В следующих строках указываются css стили.

Все стили css начинаются с новой строки и занимают одну строку:

селектор1 {правила через точку с запятой}

селектор2 {правила через точку с запятой}

Гарантируется, что внутри правил нет символов ";", но не гарантируется, что после последнего правила он есть.

Также гарантируется, что стили не содержат директив @media, @keyframes и пр.

После css-правил указывается <<<HTML>>> — признак начала содержимого html-страницы.

Гарантируется, что в html-коде нет заинлайненных стилей, тегов script и template, а текст и значения атрибутов экранированы (т.е. в них не встречаются кавычки, треугольные скобки и прочие используемые в разметке символы)

Выходные данные

Итоговый html

Пример

Входные данные	Результат работы
<pre>1 10 <<<CSS>>> #g {line-height: 1.5} .h {background: white} <<<HTML>>> <!doctype html> <html lang="en-RU"> <head> <meta charset="UTF-8"> <title>Google</title> </head> <body class="h" id="g"> <div class="h"></div> </body> </html></pre>	<pre><!doctype html> <html lang="en-RU"> <head> <meta charset="UTF-8"> <title>Google</title> </head> <body style="line-height: 1.5;background: white" class="h" id="g"> <div style="background: white" class="h"></div> </body> </html></pre>

D. Тендеры

Имя входного файла: input.txt или стандартный поток ввода

Имя выходного файла: output.txt или стандартный поток вывода

У компании ООО “Яблочко” есть несколько магазинов и один центральный склад. Каждый магазин и центральный склад вмещает определённое кол-во товаров.

Каждый день “Яблочко” проводит тендер на поставку товаров. В заявке на тендер поставщик указывает цену за единицу товара, максимальное количество единиц товара, которое он готов поставить, и магазин (или центральный склад), в который он будет поставлять товар.

“Яблочко” стремится полностью укомплектовать как можно больше МАГАЗИНОВ, потратив как можно меньше. Т.е. нехватку товаров в магазине можно покрыть товарами со склада, при этом считая, что перевозка товаров со склада в магазин поднимет стоимость этих товаров на 10%.

“Яблочко” не может заполнять склад сверх его вместимости.

Необходимо отобрать заявки и указать желаемое количество товаров по каждой отобранной заявке таким образом, чтобы получилось укомплектовать как можно больше МАГАЗИНОВ как можно дешевле.

Входные данные

Вместимость склада и магазинов по одному в строке через пробел:

<код_магазина (W - код склада, SN - коды магазинов)> <вместимость (целое число)>

Далее пустая строка и список заявок на тендер по одной в строке через пробел:

<id_заявки (строка)> <код_магазина (строка)> <цена (вещественное число)>

<макс_товаров (целое число)>

Выходные данные

Выигравшие заявки и количество товара из каждой, по одной в строке <id_заявки (строка)> <куплено_товаров (целое число)>

Пример

Входные данные	Результат работы
W 38 S1 10 S2 10 r1 S1 40 5 r2 S1 39 7 r3 S2 35 3 r4 W 30 10	r2 7 r3 3 r4 10