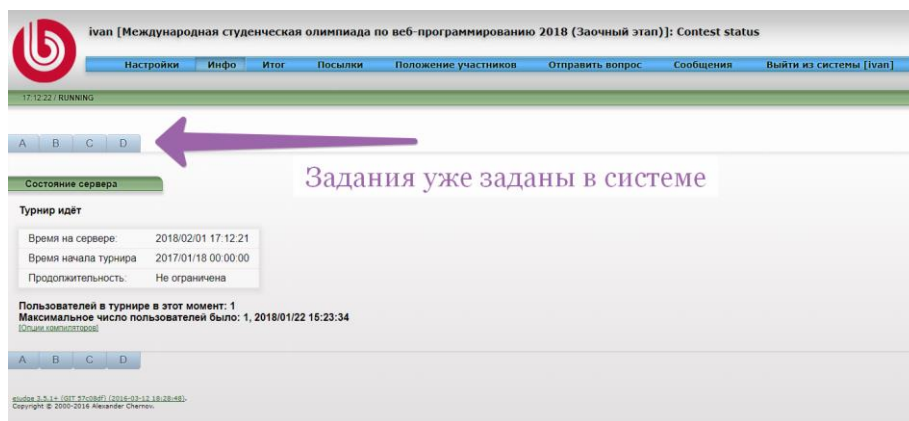


Задания заочного тура

Олимпиада по веб-программированию, Академия «1С-Битрикс», фирма «1С», 2018 год

Как решать и сдавать задания

Задания решаются на PHP. Решение заданий принимаются в автоматизированной системе по адресу <http://bitrix-olimpiada2018.ru>, необходимо пройти регистрацию. При регистрации указывайте ту же почту, что и на сайте <http://konkurs.1c.ru>. Задания уже размещены в системе.



В заочном туре четыре задания (A, B, C, D), по возрастанию сложности. Побеждает тот, кто решит больше заданий и более полно. Полнота оценивается так: при проверке решения используется несколько наборов входных данных, чем больше верных ответов – тем больше баллов.

Каждое задание можно загрузить на проверку не более 10 раз, каждая попытка уменьшает максимальное количество баллов, которые можно получить за решение.

А. По айпи вычислю

Ограничение времени: 1 с

Ограничение памяти: 256М

Имя входного файла: input.txt или стандартный поток ввода

Имя выходного файла: output.txt или стандартный поток вывода

На этой неделе фрилансер Костя познакомился с быстрорастущим интернет-магазином электроники и амбициозным интернет-маркетологом Максимом. Первая задача, которую поставил Максим удивила и одновременно обрадовала Костю. Нужно было определять город посетителя и показывать ему адрес и телефон ближайшего к нему филиала. Костя декомпозировал задачу. Для начала предстояло определять город посетителя по IP.

Входные данные

В первой строке указывается IP-адрес пользователя.

Далее указываются диапазоны IP-адресов для города, по одному в строке. Каждый диапазон описывается двумя IP-адресами через пробел и названием города.

Выходные данные

Название города, которому принадлежит адрес посетителя. Гарантируется, что указанный IP-адресу принадлежит одному диапазону из условия.

Пример

Входные данные	Результат работы
217.150.49.245 217.150.49.244 217.150.49.247 Самара 217.149.191.172 217.149.191.255 Волгоград 217.212.254.0 217.212.254.255 Москва 217.118.88.0 217.118.89.255 Саратов	Самара

В. Я календарь переверну

Ограничение времени: 1 с

Ограничение памяти: 256M

Имя входного файла: input.txt или стандартный поток ввода

Имя выходного файла: output.txt или стандартный поток вывода

Новый заказ с биржи фриланса - заказчик хочет на своём сайте показывать календарь на текущий месяц. Костя решил запрограммировать универсальный механизм, который сможет вывести календарь на произвольный месяц в произвольном году.

Входные данные

В единственной строке входных данных указывается номер месяца и год через пробел.

Выходные данные

Календарь в виде таблицы. Дни с ведущими нулями. Вместо пустых значений - два нижних подчеркивания.

Пример

Входные данные	Результат работы
1 2017	<pre>__ __ __ __ __ 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 __ __ __ __ __</pre>

С. Я за тобой наблюдаю

Ограничение времени: 2 с

Ограничение памяти: 256M

Имя входного файла: input.txt или стандартный поток ввода

Имя выходного файла: output.txt или стандартный поток вывода

2-ю неделю Костя работает над системой веб-аналитики. Эта работа заставила его посмотреть на логи веб-серверов под новым углом. Раньше он открывал их только когда сайт заказчика переставал работать. Теперь с помощью логов он должен определить, какие страницы посещал пользователь за определенный промежуток времени.

Входные данные

В первой строке указано начальное время для диапазона выборки в формате d.m.Y H:i:s.

Во второй строке указано конечное время для диапазона выборки в формате d.m.Y H:i:s.

В третьей строке указан ip-адрес пользователя (гарантируется что такой ip-адрес в логах есть) по которому необходимо построить статистику или all, если необходимо построить отчет по всем пользователям.

Далее указываются логи посещения пользователей в формате:

IP-адрес пользователя - логин пользователя при http авторизации (или "-") [дата посещения страницы в формате d/M/Y:H:i:s О - время ответа бэкенда (или "-")] код

ответа "запрос в формате: тип, url, протокол" размер ответа "URL источника запроса (или "-")" "название клиентского приложения" "-"

В веб-сервере Nginx этот формат описан следующей строкой:

```
$remote_addr - $remote_user [$time_local - $upstream_response_time] $status "$request"
$body_bytes_sent "$http_referer" "$http_user_agent" "$http_x_forwarded_for"
```

Выходные данные

Для каждого уникального IP-адреса вывести отчет в формате:

- 1) В первой строке IP-адрес.
- 2) В следующих строках вывести все страницы, которые посетил данный пользователь за указанный промежуток времени. Каждая страница описана двумя параметрами с, u, где с — количество посещений страницы, u — url страницы. Страницы выводить в отсортированном порядке по количеству посещений, если количество посещений одинаково, то сортировать в лексикографическом порядке.

Каждый новый отчет отделять от предыдущего переводом строки, выводить в порядке первого появления IP в логге.

Пример

Входные данные	Результат работы
18.01.2018 03:34:00 18.01.2018 03:35:00 213.87.137.202 68.180.228.120 - - [18/Jan/2018:03:35:00 +0300 - -] 200 "GET /spares/ HTTP/1.1" 16178 "-" "Mozilla/5.0" "-" 213.87.137.202 - - [18/Jan/2018:03:35:00 +0300 - 3.549] 200 "POST /catalog_ajax.php HTTP/1.1" 3861 "http://test.ru/" "Mozilla/5.0" "-" 213.87.137.202 - - [18/Jan/2018:03:35:01 +0300 - 2.591] 200 "POST /catalog_ajax.php HTTP/1.1" 295 "http://test.ru/" "Mozilla/5.0" "-"	213.87.137.202 1 /catalog_ajax.php

D. Быстрое решение

Ограничение времени: 6 с

Ограничение памяти: 256M

Имя входного файла: input.txt или стандартный поток ввода

Имя выходного файла: output.txt или стандартный поток вывода

Костя работает над интернет-магазином. На сайте можно купить всё от зубочисток до двигателей подводных лодок. Изюминкой сайта является умный фильтр. Он позволяет пользователям быстро находить товары с нужными свойствами и характеристиками.

Задача Кости - правильно подсчитывать количество товаров, в зависимости от галочек, которые пользователь расставил в фильтре.

Входные данные

В первой строке целое положительное число n ($1 \leq n \leq 10^4$) — количество товаров.

В каждой из следующих n строк содержится набор свойств товара в формате “ключ=значение” (от 1 до 100 свойств), разделенных одним пробелом. Каждая строка описывает один товар.

Далее указывается целое положительное число m ($1 \leq m \leq 1000$) — количество запросов.

В каждой из следующих m строк содержатся свойства запросов в формате “ключ=значение” (от 1 до 20 свойств), разделенные одним пробелом. Каждая строка описывает один запрос.

Выходные данные

Для каждого запроса вывести количество товаров удовлетворяющих условию, по одному в строке.

Пример

Входные данные	Результат работы
4	2
width=5 ht=3 len=10	2
name=circ rad=5	1
name=circ rad=5	1
name=sqr width=5 ht=3	
4	
ht=3 width=5	
name=circ	
name=sqr	
width=5 len=10	