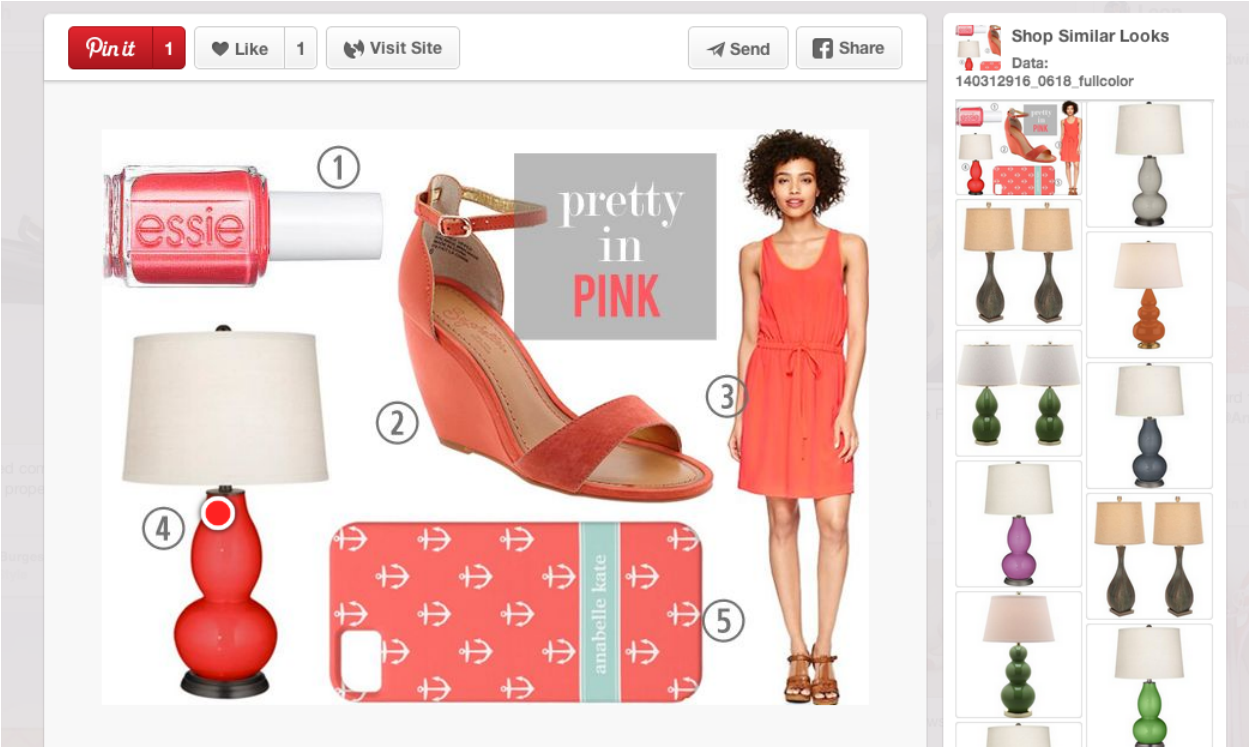
Pinterest Coding Challenge

One of the technologies we are developing at Pinterest is the ability to recognize objects within images. Your goal in this challenge will be to develop the most efficient algorithm for returning the number of images which satisfy a particular object query. Look at the image below for the visually images on the right.



Accepted Languages:
Python: Please have python2 (preferably 2.7) if you plan on using python
Java
C++

Input Files:

You are given an input file, ‘in.x.txt’ which contains object information for a set of images as well as queries. Below is a sample snippet of an input file:

4
2 1 3
2 2 3
1 3
3 4 5 6

3
1
2 and 3
(1 or 2) and 3

- The first line of the file is a number n, where n is the number of pins in the file (each line represents the objects in the pin).
- For the following n lines:
    - The first number in each line, t, is the **number of object ids** listed for a pin. For example, for the first line of pins—the line “2 1 3”—there are **2** objects associated with this pin. For the fourth pin—the line ”3 4 5 6”—there are **3 objects**
    - The subsequent t numbers are the object ids found in the pin. For the first line of pins–the line “2 1 3”–the object with id 1 and the object with id 3 are found in this pin. For the fourth pin—the line “3 4 5 6”—the objects with id 4, id 5, and id 6 are found in this pin.

- The n+2nd line (in the example case, this is “3”) is a number which represents the number of queries, q, in the file
- The following q lines then contain queries that must be satisfied.

The first query in the above snippet, for instance, asks how many pins contain the object ‘1’ (1 does because the first pin (line 2) contains the object) while the third query asks, how many images contain either objects ‘1’ or ‘2’ and object ‘3’ (2 do since the first and third pins (line 2 and 4) contain ‘1’ or ‘2’ and ‘3’). All queries will be of the form:

A_1 and A_2 and A_3...A_n

where n can be arbitrarily long. A_i where i is between 1 through n, will be either a single query or a single disjunctive. In other words, it’s either a single object, like ‘2’ or a ‘or’ of two objects, like ‘(2 or 3)’. Each A_i will contain no more than a single ‘or’.

Also note that not all queries within the ‘in.x.txt’ input file are unique.

Output File:

Please write outputs to the provided out.x.txt file name specified in the command line arguments. Output should follow the form:

1
1
2

As you can see, the above file contains one number per a query in ‘in.x.txt’. Each line should give the number of pin which match a specific query, where the order follows the same order in

which the query appears in ‘in.x.txt’. There must be a 1-1 mapping of one number for each query in order for the output to be considered correct. We can see in the above example ‘out.x.txt’ snippet for the corresponding ‘in.x.txt’ snippet that the 3rd line corresponds to the 3rd query provided in ‘in.x.txt’, since the 3rd query has 2 pins which satisfy the query.

Problem Limits:

The number of pins <= 10000
The number of objects <= 1000000
The number of queries <= 100000
The number of objects in one pin <= 100
The number of objects we want in one query <=10

Objective:

We have already provided a (very) naive implementation of the solution in solution.py that runs very very slowly **but is correct**. **Your job is to optimize the solution in solution.py** to run as fast as possible. Please make all edits in the solution.py file. You will be judged on speed, however, only submissions which return the correct result will be judged. If you are confused about any of the above you can also consult the code for clarification.

Submission Instructions:

Please do not modify the submit.py script. Doing so will result in automatic failure of the coding challenge.

Submit the script by running the submit.py script with your name, username, and language as arguments:

python submit.py <full_name> <username> <language>

For example, if your name is John Doe and your username is johndoe, submit by running the command:

python submit.py John_Doe johndoe python

Please ensure there are no spaces in either your username or your first and last name. Combine your first and last name, along with additional spaces using underscores, as in the above example, otherwise your submission will not be processed correctly. Please use the same username and name for all submissions. Submission times for the leaderboard (see below) will be shown by username, so if you want to remain anonymous for your submission, pick a username that’s also anonymous.

Language should be one of: python, java, or cpp

Submissions Leaderboard (just for fun):

To make things interesting we are keeping a leaderboard that consistently updates for the fastest solution in each language. These times will not be used to grade submission, since they are hardware dependent and can be gamed, but can be used as a reference and adds a real time competitive aspect to everything :)

Good luck and feel free to ask us any questions!