**Homework (10/4/2014)**

## 1. Ex6\_6 (codercharts.com problem“Vote Count”, use map)

## Description

The purpose of this puzzle is to count the number of votes assigned to a given set of choices and to determine the winner (the choice with the most votes) and the percentage of votes it received.  
  
If several options have the maximum number of votes they should be displayed in alphabetical order.

# Input Specifications The input file, name is data.in, is structured as follows:

1. The first line contains the number **n** of valid choices
2. The following *n* lines each contain a choice
3. The following line contains the number **p** of vote bags
4. The following *p* lines contain a bag definition **choice count** where:
   * *choice* is a choice, note that if the choice doesn't appear in the valid choices, it must be ignored
   * *count* is number of votes in that bag for that particular choice

Example:

**3**

**Coffee**

**Chocolate**

**Tea**

**5**

**Coffee 10**

**Tea 35**

**Soup 7**

**Chocolate 10**

**Coffee 25**

# Output Specifications

The output result should printed on the standard output.  
All valid options having the same maximum number of votes should be displayed in alphabetical order followed by the percentage of votes received.  
  
For the previous example, the expected output is:

**Coffee 43.75**

**Tea 43.75**

**Important:** The output percentage must have exactly 2 fractional digits, e.g. it must be of the form \*.ab where a,b are digits between 0 and 9.

## 2.Ex6\_7(codercharts.com problem“Search Trends”, use map)

## Description

[**Discuss (28 comments)**](http://codercharts.com/phpBB3/viewtopic.php?t=26)

Nowadays, every website reports their trends, or buzzwords, etc... which basically are the most frequent search requests, or post title names. The goal of this puzzle is to write a trend search which takes a list of a search requests, processes it, and returns the most frequent ones in decreasing order of popularity.

# Input Specifications

The input file, data.in, contains the search terms that were entered in the search engine. Each line contains one search term (which can be composed of one or more words).  
  
For example:

**coder charts**

**puzzle contest**

**coder charts**

**apple ipad**

**puzzle contest**

**coder charts**

**apple ipad**

**coder charts**

**puzzle contest**

# Output Specifications

Your program should compute the 10 most popular search terms and their number of occurrences, and print them from the most to the least popular. If there are less than 10 different search terms, the your program should display them all.  
The output result must be printed on the standard output.  
  
For the previous example, the expected output is:

**4 coder charts**

**3 puzzle contest**

**2 apple ipad**

Addendum: in case two terms have the same count, the order should be the ascending alphanumerical order on the terms.

## 3. Ex6\_8(codercharts.com problem“The name formatter”, us e char array, or string with string.find(…))

## Description

In pretty much every website, there are forms to fill up your name information. Users don't all capitalize their names consistently, so the website typically formats them while processing the form. The purpose of this puzzle is to format a list of names.

# Input Specifications

The input file, data.in, is structured as follows:

The first line contains the number **n** of names

1. The following *n* lines each contain a name, a name can have 1 to 3 components
   * A 1 component name is a family name, ex: CLEOPATRA
   * A 2 component name is a first name followed by a family name, ex: Albert EINSTEIN
   * A 3 component name is either a particle name (Leonardo da VINCI) or a name with a middle initial (John F. KENNEDY)
   * Example:

**5**

**CLEOpatra**

**AISHWARYA rai**

**jOHn f. KeNNeDy**

**leonardo DA Vinci**

**tyleR durdeN**

# Output Specifications

The output result should printed on the standard output.  
For each name you should print its corresponding formatted version:

* The first name must start with an uppercase letter and be followed by lowercase letters
* The middle initial must be uppercase
* The family name must be uppercase
* The particle must be lower (the da in da VINCI)

For the previous example, the expected output is:

**CLEOPATRA**

**Aishwarya RAI**

**John F. KENNEDY**

**Leonardo da VINCI**

**Tyler DURDEN**

4. Keep working on USACO training problems in 1.2.

When one problem cannot be solved in 2 days, let me know that I’ll give solution to move forward.