

Computer Science II — Homework 4 — Mail Distribution

Overview

This assignment is due Thursday, February 21 at 11:59:59pm and is worth 75 points toward your homework grade.

Due to an increase in the volume of mail, the local post office needs an automatic mail sorting mechanism. You have been volunteered to write a software system to help do this.

Input

The input to the program consists of two files. The first contains the mail delivery routes. Each route has a name and a series of street names (single strings). The order of these street names is important because this is the order in which the carrier will deliver the mail. The format of the mail-delivery-route input files is as follows. Each route will be specified by a name (a single string), an integer n giving the number of streets on the route, and a sequence of n strings specifying the streets on the route. After the n strings for a route have been read, if there is another string in the file, then there is another route. Otherwise input should be terminated.

The second input file contains a mixture of two different types of input. Both are on a single line of input and both involve the input of an integer and two strings. An example illustrating the first is

```
1600 Pennsylvania hdy78324hsdf
```

which says that there is a new mail item with unique id `hdy78324hsdf` to be delivered to building number 1600 on street `Pennsylvania`. (Note the route name is not specified here.) An example illustrating the second is

```
1600 Pennsylvania pick-up
```

which says that the occupant of building number 1600 on street `Pennsylvania` is picking up his/her mail.

The two input files will not have any mistakes: there will be no formatting errors in the input files (such as the wrong number of street names for a route); no streets will be repeated; each street in the mail-items file will correspond to a street on one of the routes; no mail ids will be repeated.

Output

Your program should only have two types of outputs. The first is the output in response for mail pick-ups. In this case your program should output the address and the ids of the mail items, one per line. The order of the items should be order in which the items were input to the systems. The second output will show the assignment of mail items — the ones that were not picked — up to the mail routes. The routes and the streets within the routes should be ordered as they are ordered in the input file. For each street, the order of output should be by increasing street number. For items going to the same address, the order should be the order they were read into the system rather than the id number order.

See the example output for details on the output format.

Requirements

This program must use lists and iterators. It may not use vectors and it may not use techniques beyond Lecture 9.

The process of searching for the appropriate location for a mail item is the heart of the program, and you must make an attempt to make this reasonably efficient (although the requirement to use lists does place a limit on efficiency for this particular problem). In particular, you may not search the entire list of mail items assigned to a route in order to insert a mail item — at most you may search the list of mail items for a particular street.

Submission

Please place all of your source code files and `readme.txt` file in a folder named `hw4` (no spaces, no uppercase, not `hw4_submit`, not `HW4`). Make sure none of your file names include spaces. Zip your folder prior to submission, and submit it through course web page.