

## Lab 7

### **Problem Description:**

Write a Java program that receives multiple lines of string blocks, and prints out all possible strings which can be made by concatenating exactly one string block from each line together, with the condition that the concatenated strings must contain our target string or some rotation of our target string. E.g. If our target string is “mytarget”, then “mytargetap” and “xltmytarge” are both acceptable combinations, while “rgetcmymta” and “ymytargetap” are not.

**Input:** The first line of the input is an integer N ( $1 < N < 5$ ) determining how many lines of strings we will have. In the following N lines, each line will contain between 1 and 10 comma-separated “string blocks”, E.g. in the second line of test case 1 below, “kdzmy” is a string block. Each block is a string of 1 or more lowercase letters. The last line defines the target string that we are looking for.

**Output:** Print out all possible combinations of exactly one block from each line which contain our target string or some rotation of our target string. E.g. rotations of the target string “mytarget” would be: “ytargetm”, “targetmy”, “argetmyt”, and so on. Output strings should be sorted in alphabetical order. If several different unique combinations all yield the same acceptable string, then print that string multiple times (see test case 2 below).

| Test Case | Input   | Output   |
|-----------|---|--|
| 1         | 3<br>kdzmy,myta,etmy,ytarg<br>ym,r,ta,yta,xltm,etc<br>getap,rg,e,tarz<br>mytarget | etmytarg<br>mytargetap<br>mytargetc<br>rgetmyta<br>targetmy<br>xltmytarge<br>ytargetmy   |
| 2         | 3<br>vi,vifoo,t,i<br>o,olefoo,le<br>let,vio,tv<br>violet                          | letvio<br>letvio<br>letviolefoo<br>olefooletvi<br>olefooletvifoo<br>oletvi<br>oletvifoo<br>tviole<br>tviolefoo<br>violet<br>violet |