**6. Naughty List**

# Program Name: Naughty.java Input File: naughty.dat

Sammy Klaws, the eternal spirit of Christmas, has returned to continue the annual tradition of giving the good students their presents and the bad students their coal. Usually, this would not be a difficult task, since Sammy does the same thing every year and has a very efficient system in place. However, Sammy practiced a little too much computer science this year, and his vision has gotten a lot worse to the point where he sometimes cannot distinguish similar-looking names. This is a problem, because a lot of kids on his naughty list have similar names to the kids on his nice list, so he’s enlisted you to write a program to see if the names are actually right. Given the correct name and a list of other names that Sammy thinks are the same but may be slightly different, find the name that has the most correct characters in the same spot of the correct name. If no character is correctly placed, a failed message will be printed. If more than one name has the same number of characters in their correct places, the name that appeared first will be printed. For each test case, print out the most similar name, followed by an accuracy percentage that represents the percentage of letters that were in the correct position using this equation:

The length of each string will never exceed 100.

**Input**The first line will contain integer n*,* which represents the amount of test cases to follow. The next ndatasets will each start with integer s, followed by the correct string, and s-1 lines, each with a random permutation of the original string.

**Output**

Output the string that has the greatest number of corresponding letters, and on the next line, print “Accuracy Percentage: ’’, followed by the accuracy percentage calculated using the formula aforementioned and rounded to a whole number. If no strings were found to have even one character in the corresponding place, print “FAILED” on a new line instead of the corresponding string and for the accuracy percentage, print “NaN” instead of a percentage.

**Example Input File**

2

4

wildcat

wdatlci

awtcdil

wiadclt

5

squid

qdisu

qsidu

diqus

usqdi

**Example Output to Screen**

wiadclt

Accuracy Percentage: 71%

FAILED

Accuracy Percentage: NaN