Two distinct words can be joined if one, two or three of the letters at the end of the first word are the same as the letters at the beginning of the second word, and only if the letter/letters are in the same order.

**FOR EXAMPLE:**

**Apple** and **pleasant** can be joined to form **appleasant**.

**Grill** and **lamma** can be joined to form **grillama**.

**Predicament** and **tension** can be joined to form **predicamentatension**.

A word-snake of **length** **n** is a sequence of **n** words that can be joined each to the next to form one long string. Write a Java program that prompts the user to enter the name of the data file whose first record is a positive integer **n(n<30)** followed by **n** lines each containing a single word. The words consist only of **lower-case letters**, and are **at least 4 letters long** and are **in alphabetical order**.

Treat the problem as a **directed graph** with **n** nodes, where **nodes represent** words, and an edge from node **i** to node **j** means that word **i** and word **j** can be joined. **A word cannot be adjacent to itself**, even a word like outshout. Your program should prompt the user to enter two words (in lower case) at the keyboard and then write to the screen a single line of output something like one of the following:

**You entered a word that is not in the list.**

(Stop the program if this happens.)

**The words “apple” and “telephone” are not connected.**

(Print this if there is no word-snake beginning with “apple” and ending with “telephone”.)

**The shortest snake that begins with “apple” and ends with “telephone” has length 52.**

(Print this if “apple” is connected to “telephone”, and the shortest path connecting them is of length 51. You do not have to give the words that make up the word-snake.)

You will need a method that determines whether two words can be joined or not. This is a directed graph, so word **i** may be adjacent to word **j** without word **j** being adjacent to word **i**. There are pairs of words (like **chair** and **ironic**) where each is adjacent to the other, but this rarely happens.

Sometimes words can be joined in more than 1 way, like “phenomenon” and “nonsense”. These two words can be joined by the single letter ‘**n**’ or by the substring”‘**non**”. In such cases the only thing that matters is that they can be joined. As stated above, in this problem we do not allow a word to be joined to itself even if it begins and ends with the same letter(s), such as “chronic” or “sturdiest”. However, we do allow a word to be connected to itself, i.e. a word-snake of length 3 or more may begin and end with the same word, as seen by the following word-snake of length 5:

**EXAMPLE:**

“keyboard” -> “dragon” -> “onset” -> “turkey” -> “keyboard”

**Adjacency Matrix**

10 by 10 matricies

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |
|  | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
|  | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
|  | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
|  | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 4 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
|  | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 |
|  | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 |
|  | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
|  | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
|  | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |

**From Data File Entries of: Here are all of the answers of the shortest possible path**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |
|  | 4 | 3 | 2 | 2 | 3 | 5 | 1 | 2 | 4 | 4 |
|  | 5 | 4 | 4 | 3 | 4 | 2 | 3 | 4 | 1 | 1 |
|  | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
|  | 2 | 1 | 4 | 4 | 1 | 3 | 3 | 4 | 2 | 2 |
|  | 1 | 4 | 3 | 3 | 4 | 6 | 2 | 3 | 5 | 5 |
|  | 3 | 2 | 2 | 1 | 2 | 4 | 1 | 2 | 3 | 3 |
|  | 3 | 2 | 1 | 1 | 2 | 4 | 4 | 1 | 3 | 3 |
|  | 3 | 2 | 1 | 1 | 2 | 4 | 4 | 5 | 3 | 3 |
|  | 4 | 3 | 3 | 2 | 3 | 1 | 2 | 3 | 4 | 4 |
|  | 5 | 4 | 4 | 3 | 4 | 2 | 3 | 4 | 1 | 5 |

10

amateur

candy

edict

erratic

icecream

lower

release

sensible

yellow

yesterday

10 by 10

When reading the data file make sure the 1st entry is a positive integer smaller than 30, if not return the prompt to the user asking for a valid data file

When reading the data file, read them in to Lower so all the letters are in lower case

When reading the data file make sure all of the words are at least 4 characters long, if they are not print an error message to the screen saying that all of the words must be at least 4 characters long and to please chose a correct data file returning the prompt to the user.

If there is a path between the 2 words chosen by the user also print the list of words that make up the path with a space between them.

EX: “The words apple and telephone are connected. The path is: apple candy telephone.”