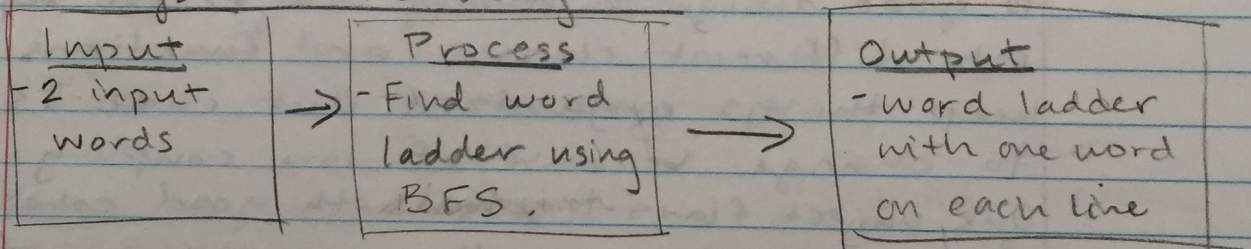
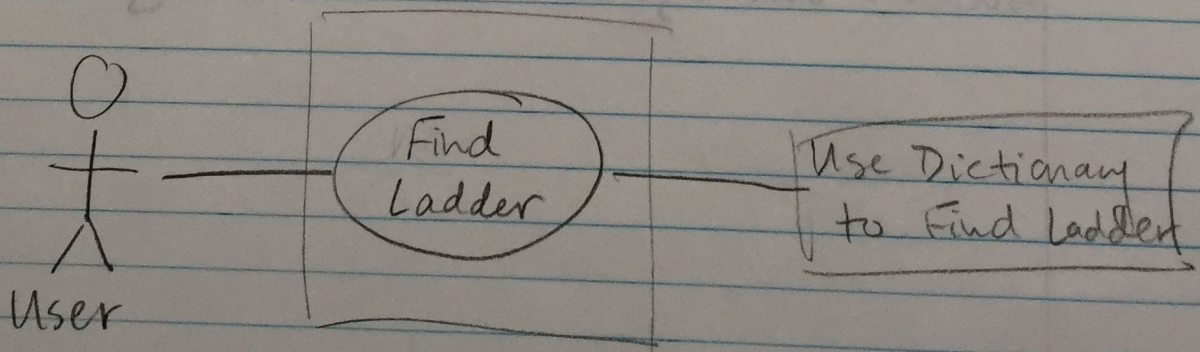


Analysis: We are trying to create a word ladder which takes 2 input words. The first word is the top of the ladder and the second word is the bottom. We are using a dictionary to find words to connect the two input words, by only changing one letter at a time. We are assuming that the word ladder connects the words, not that it is the shortest possible. We are not going to change the same letter position twice in a row. Also we won't use the same word twice in our word ladder. Is there a limit on the maximum word ladder length?

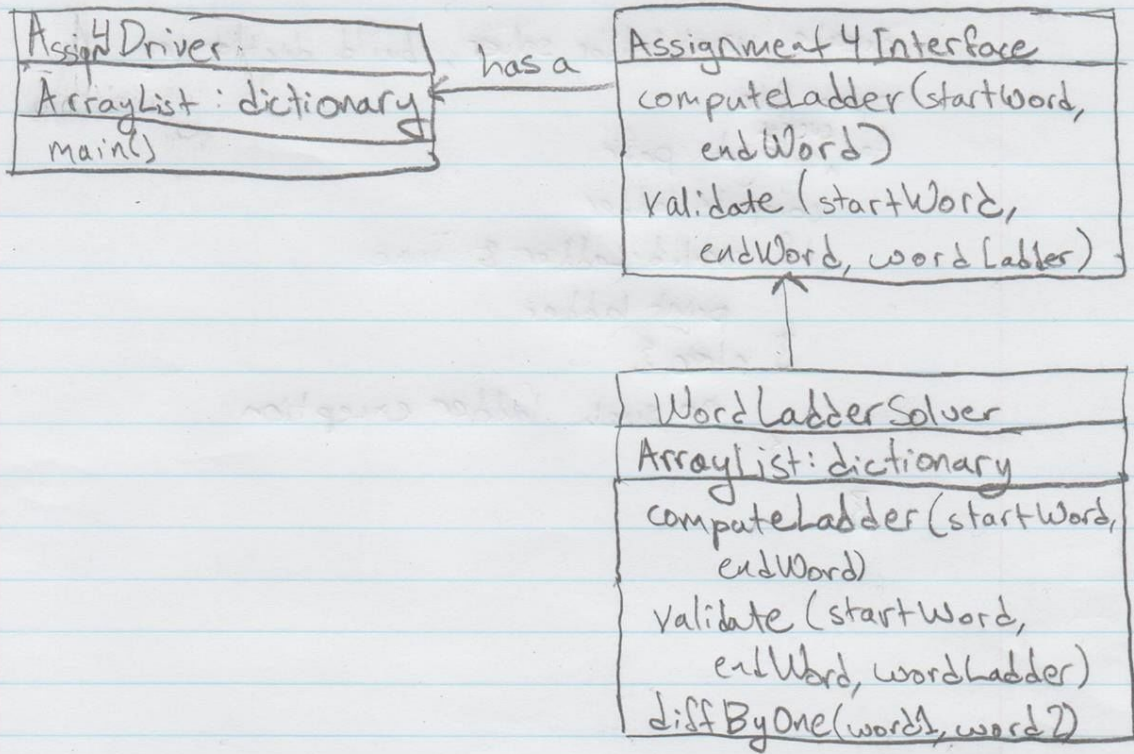
A System IPO diagram



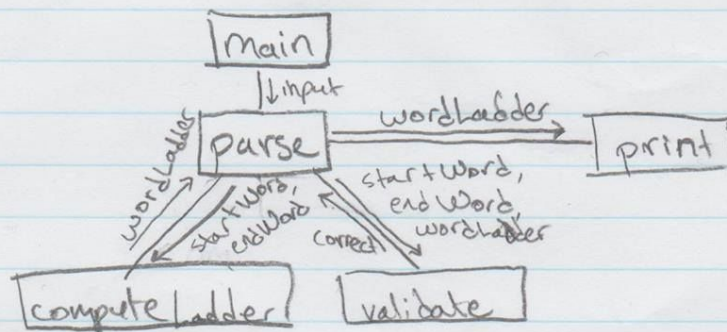
A Use - Case Diagram



UML Diagram



Functional Block Diagram



Algorithm

create word ladder solver, build dictionary

open file

for every word pair

compute ladder

if valid ladder Σ

print ladder

} else Σ

no such ladder exception

}

}

Paragraph: Our OOD reflects the interaction and behavior of the real-world objects that it models. The dictionary is a great example. It is a list of words in order and allows you to determine if a word is valid, just like in the real world. The PDF gave the algorithm for a recursive DFS method. We actually decided to complete the lab using BFS. DFS just gives you the first ladder it finds, while BFS gives you the shortest ladder. From a programming perspective, DFS uses a stack and BFS uses a queue. Our design could be easily modified to allow for a bigger dictionary with words more or less than 5 letters. Our design adheres to good design in many ways. We separate into different classes and functions to keep things organized. Our code is easy to change because of low coupling. Our project flows together with good cohesion. The user only sees the necessary info such as the word ladder, or statements like invalid input or no such ladder, which shows our good info hiding.