1. Consider the problem of constructing crossword puzzles (not solving): fitting words into horizontal and vertical squares. Assume that a list of words (dictionary) is available and the task is to fill in the squares using and subset on this list. Go through the goal and problem formulation and choose a strategy to solve it. Specify heuristic function if you think one is needed. Suppose that black and white squares are already predefined, i.e. slots for the words are defined

- Initial state: A grid is empty and dictionary has all the words.
- Goal: Every word is filled with a different word in every pair of intersecting words agrees on the intersecting letter.
- Successor function: Pick a word slot that has not been picked already and fill it with any word from the dictionary that agrees with any letters in place. You need to also pick the words slots which have not been picked but are already full with letters, such that you can check whether those letters actually make words.
- Path cost: each operation has cost one since the goal is the same for all. This is a CSP. We can use backtracking to solve it. Each variable is a slot and the values are the words of the correct length for the slot.
- Heuristic: MRV would work best. MRV would pick the longest words slots first - because there are more short words than long words. LCV may also work well but it is expensive (quadratic in the size of dictionary with simple implementation). It can be approximated by preferring words such that the letters which intersect other words slot are commonly encountered letters.

2. Exercise 6.1 in AIMA book. The diagram with the evaluation function below the terminal nodes is below. The backed up values are next to the nonterminal nodes. The nodes enclosed in squares are the ones which don’t need to be evaluated assuming optimal ordering.