1. In the backtracking algorithm for SAT, suppose that we choose a subproblem (CNF) formula that has a clause that is as small as possible; and we expand it along a variable that appears in this small clause. Show that if the input formula only contains clauses with two literals (that is, it is an instance of 2SAT), then a satisfying assignment, if one exists, will be found in polynomial time.

2. Devise and implement (in your favorite programming language) a branch-and-bound algorithm for the SET COVER problem:

   *Input:* A universe $U$ of $n$ elements, a collection of subsets of $U$, $S = \{S_1, \ldots, S_k\}$

   *Output:* A subcollection of $S$ that covers all elements of $U$

   *Cost:* Number of subsets

   (a) What is a subproblem?

   (b) How do you choose a subproblem to expand?

   (c) How do you expand a subproblem?

   (d) What is an appropriate lowerbound?

3. In the MULTIWAY CUT problem, the input is an undirected graph $G = (V, E)$ and a set of terminals $S = \{s_1, s_2, \ldots, s_k\} \subseteq V$. A multiway cut is a set of edges whose removal disconnects the terminals from each other. The multiway cut problem asks for the smallest such set.

   (a) Show that this problem can be solved exactly in polynomial time when $k = 2$.

   (b) Design and implement a local search algorithm for multiway cut.

4. (Bonus) Design and implement a backtracking algorithm for solving SUDOKU puzzles.