This homework covers designing backtracking algorithms. It is due on Wednesday, April 22. Note that there is no class that day — you can hand your writeup in during office hours or put it under my office door. Please do not email your homework.

See the Policies page on the course website for information about revise-and-resubmit, late work, and academic integrity as it applies to homework.

Write your solutions carefully — your work should be neat, readable, and organized. Keep in mind that what you hand in should be a presentation of your work, not your train-of-thought stratch work with a solution mixed in somewhere.

Develop backtracking algorithms using the process discussed in class for #1–3. This is not a research task — the point here is to understand and be able to apply the process discussed in class to develop an algorithm yourself.

Show your work — and your understanding of the steps in the process — by including each of the steps in your writeup. Don't just give an algorithm! See the posted writeups from class for examples. (Don't include the marked "commentary" parts.)

1. Using just four colors, color a (geographical) map so that no two bordering regions have the same color — or determine that it isn't possible. The input to this problem can be represented as a graph — each region is represented by a vertex and bordering regions are represented by edges between the respective vertices.

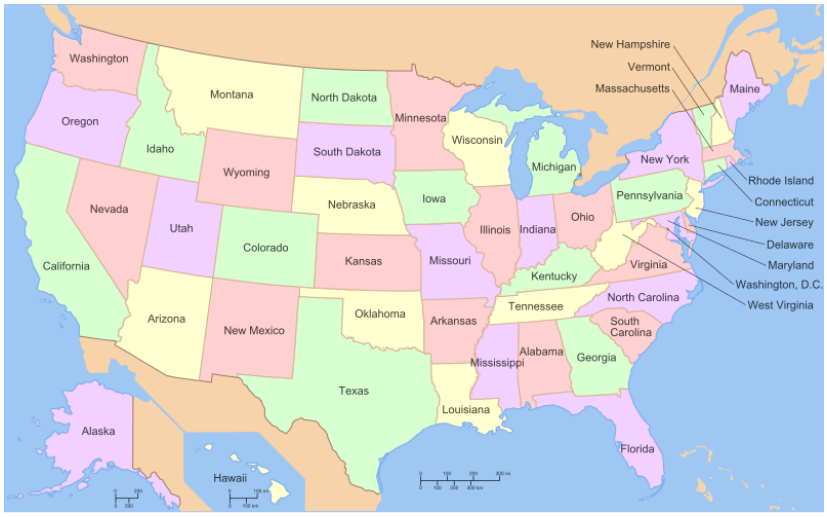

image credit: https://commons.wikimedia.org/wiki/File:Map_of_USA_with_state_names_2.svg

2. You are planning a dinner party, with n guests to be seated at two tables. (Each table holds exactly half of the guests.) You also have a list of preferences — some pairs of guests would prefer to be seated at the same table, while other pairs prefer to be at different tables. (There may also be many pairs with no preference about tables.) List all the possible seating arrangements that respect the guests' preferences. (The order of seating around each table doesn't matter, just who is at which table.)

3. You have a set of n jobs to complete, and n contractors have each bid on each of the jobs. Let $c(i, j)$ be the cost of having contractor j complete job i. Find an assignment of jobs to contractors so that each job is assigned to one contractor, each contractor is assigned one job, and the total cost of getting all of the jobs done is minimized.

4. Do the homework #11 drill problems on Canvas. (Look for `hw11 drill` in the Quizzes section.)