

CMSC 671 – HW 2, Fall 2024

Turnin: Blackboard.

Please submit all parts as a **single PDF file** named *yourlastname_hw2.pdf*.

All files must start with your last name and have your full name **in the file**, at/near the top.

PART I. CONSTRAINT SATISFACTION

For these questions, you will be solving a map-coloring problem on the map shown here. Each of the regions A–E must be green, blue, red, or yellow, and no adjacent regions may be the same color.

Backtracking search

You will use backtracking search to find a solution to this constraint satisfaction problem. The **variable ordering heuristic** is to make assignments to variables in **minimum remaining value** order, breaking ties **alphabetically by variable name**; the **value ordering heuristic** is to consider the values in the order blue, green, red, yellow.

For each node, indicate all variables with, e.g., $A=r$, $B=g$, etc. Uninstantiated variables can be shown as, e.g., $A=$. (See Figure 2.) At each layer of the search tree, show nodes for the variable you are assigning a value to.

Assignment: (10 pts)

- 1) Show the search tree created by following this algorithm, and circle the solution node.

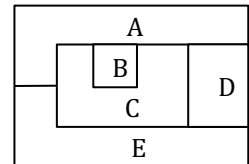


Figure 1. Color this map blue, green, red, and yellow with no adjacent areas sharing a color.

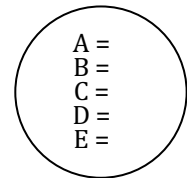


Figure 2. Start node.

Scheduling

You have been asked to schedule seven computer science professors to give talks in your department. Unfortunately, you discover that there are only four available times, meaning that some of the talks will necessarily overlap. To make things more complicated, various professors insist on being able to see certain talks, which means you have to figure out how to schedule the talks such that nobody is disappointed. The following are all the constraints you find when trying to create the schedule:

The list of guest lecturers consists of Drs. Baker, Carter, Daniels, Edwards, Fisher, Garcia, and Hall. The available dates are October 1st, 2nd, 3rd, and 4th.

- Hall can only speak on the 2nd.
- Daniels and Carter want to see each other's talks and so cannot overlap.
- Your advisor wants to see all of Daniels, Fisher, and Edwards, so they cannot overlap.
- Baker and Edwards have been competing to be first to prove that $P=NP$ and all dislike each other. They must be scheduled for different days so they don't run into each other.
- Hall is working on fundamentals and Baker is working on applications, so Hall should speak before Baker.

- Garcia and Fisher cannot be scheduled for the same day, as a particular professor wants to take them each to lunch.
- Carter and Fisher split an EasyPass and cannot travel on the same day, so they cannot overlap.

Assignment: (20 pts)

2) Give a CSP formulation for this problem:

- What are the *random variables*? (There are multiple options; choose carefully.)
- What is the *domain of each variable*?
- For all variables, express the constraints using the domain. Spell out constraints—do not just state the constraints in English sentences.

3) Draw a constraint network that captures all the constraints. Include labels on nodes and edges.

4) Give a full legal instantiation for this problem, or explain why none exists.

5) Explain how you found the solution. What search approach did you use? How did you order variables and values? Did you do constraint propagation? Forward checking? Give details.

PART II. BAYES' NETS AND PROBABILITY

Consider the Bayesian belief network shown in Figure 3, with the following conditional probabilities:¹

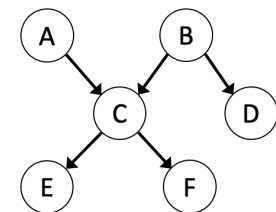


Figure 3. A six-variable Belief Network with associated CPTs.

$P(a) = 0.9$	$P(d b) = 0.1$
	$P(d \neg b) = 0.8$
$P(b) = 0.2$	
$P(c a,b) = 0.1$	$P(e c) = 0.7$
$P(c a,\neg b) = 0.8$	$P(e \neg c) = 0.2$
$P(c \neg a,b) = 0.7$	$P(f c) = 0.2$
$P(c \neg a,\neg b) = 0.4$	$P(f \neg c) = 0.9$

Assignment: (25 pts)

6) Compute $P(e)$ using variable elimination. You should first prune irrelevant variables. Show the factors that are created for a given elimination ordering.

¹ Question drawn from Mackworth & Poole, Artificial Intelligence: Foundations of Computational Agents, 2nd Edition.

PART III. GAME PLAYING: FLIPPING A COIN

Consider a two-player coin-flipping game where two players alternate flipping a two-sided coin. The rules are as follows:

- If the coin lands heads up, the player who flipped gains two points.
- If the coin lands tails up, the opponent gains one point.
- If a player exceeds four points, they automatically lose all of their points, and the game ends.
- On their turn, a player can choose to stop the game, in which case both players keep their current scores.
- The goal is to beat the other player **by as many points as possible**.

Assignment: (15 points)

7) Given the expectiminimax algorithm:

- a) Draw the 4-ply (two moves for each player) expectiminimax tree for this problem.
- b) On this tree, using the static evaluation function ($\text{score}(\text{player1}) - \text{score}(\text{player2})$), back up the leaf values to the root of the tree.

8) Consider the actions taken by each player:

- a) What is the best first action for the first player to take? (Play or stop?)
- b) If player 1 flips tails, what should player 2 do? Why? (1-3 sentences)
- c) Would you describe this game as fair? Why or why not? (1-3 sentences)