

COMP3702/7702 2020 - Mock Exam

MODULE 1: SEARCH

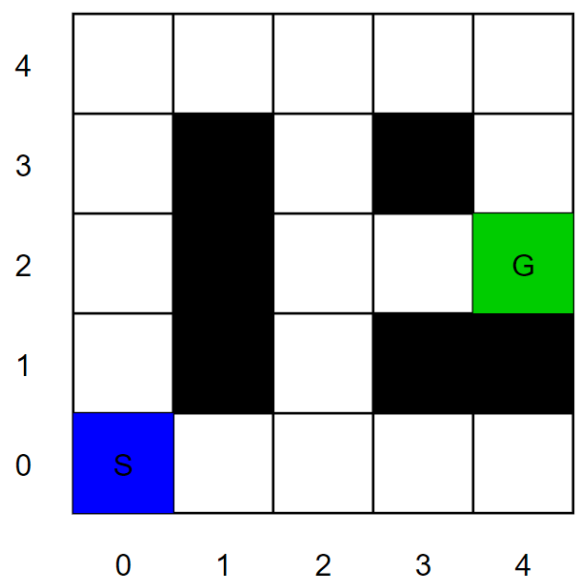
Question 1.1

Regarding blind search algorithms:

- What data structure is best used to implement breadth-first-search?
- What blind search algorithm or algorithms can be implemented using priority queue?
- Explain in your own words the benefits of using iterative-deepening depth-first search over depth-first search in a search problem.

Question 1.2

Answer the following questions about the search problem on the gridworld shown below:



The details of this problem are:

- The 5x5 grid has obstacles indicated by black tiles in three contiguous blocks, at coordinates (1,1), (1,2), (1,3); at (3,1) and (4,1); and at (3,3).
- The initial state is S, at coordinates (0,0), and the goal state is G, at (4,2).

For the questions that ask for a path, please give your answers in the form 'S-(x,y)-...-(x,y)-G'. Break any priority ties using the move order *up, right, down, left*.

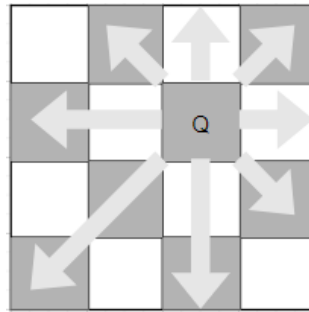
- What path does iterative-deepening depth-first search with depth parameter $k = 3$ return for this search problem?
- Now assume a constant cost of -1 per move. What path does uniform cost search return for this search problem?
- Consider a heuristic for this search problem given by $h = 4 - x$ (x is the x-coordinate).
 - Is the heuristic h consistent?
 - Is the heuristic h admissible?

MODULE 2: REASONING WITH CERTAINTY

Question 2.1

In chess, a queen can move in straight lines any number of squares left or right, forward or backward, or diagonally. The 8-queens puzzle is a classic problem of placing eight chess queens on an 8x8 chessboard so that no two queens threaten each other.

A simpler version of the problem is the 4 queens puzzle, played on a 4x4 board, shown below:



The possible moves of a single queen are shown in this figure.

The typical way to model this problem is to assign each of the 4 queens its own column, A, B, C or D, and then choose a row 1, 2, 3 or 4 in such a way that they cannot attack each other. Starting from this variable definition, answer the following questions on the 4 queens puzzle.

- What is the domain of each queen?
- List all binary constraints between variables for this CSP. How many constraints are there?
- Express the problem as one of logical validity in *conjunctive normal form*.
- Assume a partial assignment is given, where Q-A is placed in row 3. Apply *backtracking search* starting from this partially-assigned CSP. Use the variable ordering (Q-B, Q-C, Q-D) and the variable domain order 1,2,3,4 to expand nodes in the search graph. List all variable assignment and removal operations, and any backtracking operations.
- Give a solution to this CSP.

Question 2.2

- Construct a truth table to show that $\neg(p \vee q)$ is logically equivalent to $(\neg p \wedge \neg q)$.
- Given the premises $(p \Rightarrow q)$ and $(r \Rightarrow s)$, use the Propositional Resolution rule to prove the conclusion $(p \vee r \Rightarrow q \vee s)$.

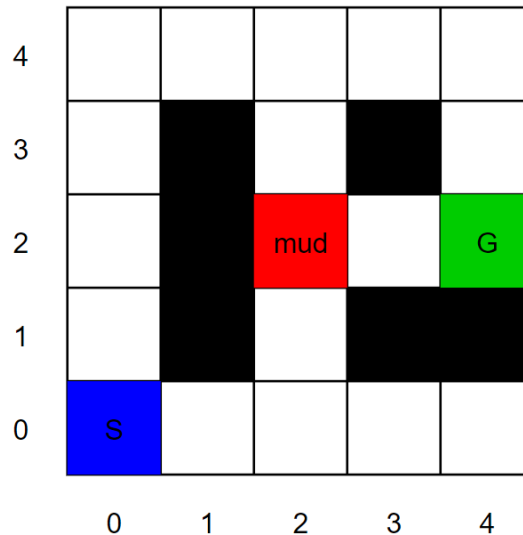
MODULE 3: DECISION MAKING UNDER UNCERTAINTY

Question 3.1

The *decomposability* axiom of the rational preferences utility model asserts that “there is no fun in gambling.” Explain how an agent whose preferences violate the decomposability axiom can be manipulated using the concept of a *money pump*.

Question 3.2

Answer the following questions about the gridworld Markov decision process shown below:



The details are:

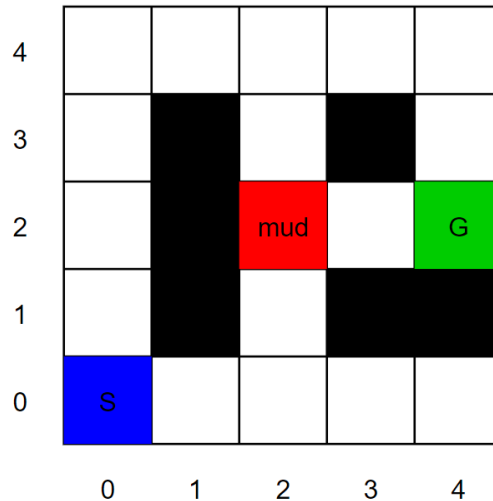
- The 5x5 grid has obstacles indicated by black tiles in three contiguous blocks, at coordinates (1,1), (1,2), (1,3); at (3,1) and (4,1); and at (3,3).
 - The agent starts at the state labelled S, at coordinates (0,0).
 - Its goal state is labelled G, at (4,2).
 - Entering the goal state earns the agent 10 points.
 - Actions that cause collisions with the boundary or obstacles keep the agent at its current state (with no cost collision).
 - The red square at (2,2) is mud, which can slow the agent down. The agent successfully moves out of the red square with probability 0.2, and remains stuck in the mud with probability 0.8.
 - Assume $\gamma = 0.8$.
- a) What is the transition function for each action starting in the mud? That is, write out $T(s,a,s')$ for each s' with non-zero transition probability, starting at $s=(2,2)$.
- b) Initialise all values to 0, and compute three iterations of (synchronous) *value iteration* for the gridworld above. What are the value function iterates (i.e. approximate values) for each state after:
- i. the first iteration,
 - ii. the second iteration, and
 - iii. the third iteration.

List only the non-zero iterates; all unstated values will be assumed to be equal to 0.

MODULE 4: LEARNING TO ACT

Question 4.1

Using the same gridworld as above, assume now you do not know anything about the transitions or rewards. The gridworld is shown again below:



You choose to use SARSA to solve this problem, and employ an epsilon-greedy exploration strategy with exploit probability $\epsilon = 0.8$ and exploration using uniform random sampling of any action otherwise.

Suppose you obtain the following observations:

s_t	a_t	s_{t+1}	reward	a_{t+1}
(2,1)	Up	(2,2)	0	Up
(2,2)	Up	(2,2)	0	Right
(2,2)	Right	(3,2)	0	Right
(3,2)	Right	(4,2)	10	Restart

- What values does the Q-function attain if we initialise the Q-values to 0 and replay the experience in the table exactly two times? Use a learning rate $\alpha = 0.6$, and discount factor $\gamma = 0.8$. List only the non-zero approximate Q-values; all unstated Q-values are assumed to be equal to 0.
- Using these Q-values and the epsilon-greedy exploration strategy describe above, what are the probabilities of taking each action next time the SARSA agent gets stuck in the mud?

MODULE 5: REASONING ABOUT OTHER AGENTS

Note: The questions below will be answered in the Week 12 Lecture

Question 5.1

In the game of *Morra*, each player shows either one or two fingers and announces a number between 2 and 4. If a player's number is equal to the sum of the number of fingers shown, then her opponent must pay her that many dollars. The payoff is the net transfer, so that both players earn zero if both or neither guess the correct number of fingers shown.

In this game, each player has 6 strategies: she may show one finger and guess 2; she may show one finger and guess 3; she may show one finger and guess 4; or she may show two fingers and guess one of the three numbers.

- a) There are two *weakly dominated strategies* in *Morra*. What are they?
- b) Imagine that player A can read player B's mind and guess how he plays before he makes his move. What *pure strategy* should player B use?
- c) Player B consults a textbook and decides to use randomisation to improve his performance in *Morra*. Ideally, if he can find a best mixed strategy to play, what would be his expected payoff?
- d) One possible mixed strategy is to play show one finger and call "three" with probability 0.6, and to show two fingers and call "three" with probability 0.4 (and play the other strategies with probability 0). Is this a Nash equilibrium strategy? Assume that Player B is risk neutral with respect to the game payoffs.