# Com S 472/572 Principles of Artificial Intelligence

## Midterm Exam

Fall 2020

Honor statement: *I affirm that I am the assigned student taking the test, and this is entirely my own work. I affirm my acceptance of these rules: 1) closed-book and closed-notes during the exam; 2) no online search for information during the exam; and 3) no discussion or sharing in any form with others during or after the exam.*

Name: ______Sean Gordon_________

ISU NetID (username): _____________Sgordon4__________________@iastate.edu

| 1 | 2 | 3 | 4 | 5 | Total |
|---|---|---|---|---|---|
| | | | | | |
| 34 | 12 | 22 | 16 | 16 | 100 |

1. **[34 pts]** *Short Questions*

(a) [**12 pts**] Determine if the following statements are true or false. For each statement, mark only the answer you think is correct.

(i) The A* search algorithm using an admissible heuristic function $f$ will never expand more than node $n$ with $f(n) = C^*$, where $C^*$ is the cost of the optimal solution path.

true __________ false X

(ii) A simulated annealing algorithm is more likely to tolerate bad moves at the start than later.

true X false _________

(iii) Not every constraint satisfaction problem (CSP) can be transformed into a CSP with only binary constraints.

true __________ false X

(iv) The program AlphaGo employed Monte Carlo tree search to defeat the world no. 1 ranking Go player in 2017.

true X false _________

(v) A hill climbing search in the vicinity of a local maximum state is capable of moving past it to continue the search process.

true ___________ false X

(vi) A propositional sentence $\alpha$ entails another sentence $\beta$ whenever the model set of $\beta$ is a subset of the model set of $\alpha$.

true X false __________

(b) [**6 pts**] Fill answers (Yes / No) in the following table which compares uninformed search algorithms. Assumptions: (i) the branching factor is finite, (ii) the state space either has a solution or is finite, and (iii) all action costs are identical.

| | Breadth-First | Depth-First | Iterative Deepening |
|---|---|---|---|
| Complete? | Yes | Yes | Yes |
| Optimal cost? | Yes | No | Yes |

(c) [**3 pts**] The branching factor for the state-space graph of the 8-queens problem is **8**
(Make it as tight as you can.)

(d) [**3 pts**] If a problem has $n$ physical states, its number of belief states is **2^n**
(Give the exact expression. No big-$O$ notation.)

(e) [**4 pts**] A depth-limited search employs a depth limit $d$ to avoid exploring an infinite path. Suppose that the search graph has branching factor $b$. In big-$O$ notation give the complexities of time and space for the search.

Time complexity: $O(\_b\text{^}d\_)$

Space complexity: $O(\ d\ )$

(f) [**6 pts**] Explain how a local beam search works.

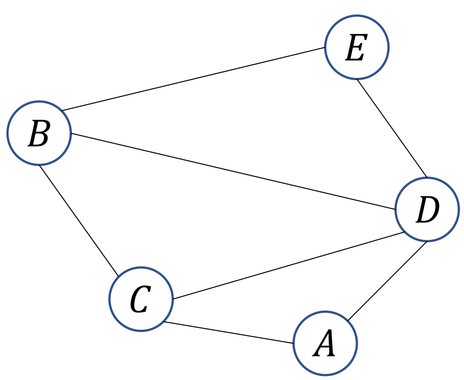
Local beam search begins by randomly generating k states, then continues by generating each state's successors and keeping the k best. The algorithm then stops if one of the states is a goal state.

2. **[12 pts]** *Constraint Satisfaction*

(a) [**6 pts**] Consider the constraint $Y = \sqrt[3]{X}$, where $X, Y$ are integers such that $0 \leq X, Y \leq 100$. Give the new domains of $X$ and $Y$ after making the two variables arc-consistent.

Y={0, …, 100}, X={-1000000, …, 0}

(b) [**6 pts**] Suppose you are given a graph $G$ and asked to color its vertices in Red, Green, or Blue. (Assume that the graph can be colored this way.) If $G$ is the graph shown below, which vertex would you color first? Explain the reason for your choice. In general, what vertex in $G$ would you first choose to color?




I would color vertex D first, as it is the vertex with the largest degree. This would minimize the amount of possible future moves.

3. **[22 pts]** *A* search and the 8-puzzle*

| | | |
|---|---|---|
| 1 | 2 | 3 |
| 8 | | 4 |
| 7 | 6 | 5 |

Goal state $S_g$

The 8-puzzle in this problem has its goal state $S_g$ shown on the left. To solve an 8-puzzle, we have two heuristic functions: $h_1(S)$, which counts the number of misplaced tiles in a state $S$ when compared to the goal state $S_g$, and $h_2(S)$, which sums up the Manhattan distance between the position of every tile in $S$ and its position in $S_g$.

(a) [**6 pts**] Evaluate the two heuristics on a state $S_0$ shown below on the left.

| | | |
|---|---|---|
| 2 | 3 | 1 |
| 6 | 5 | 7 |
| 8 | 4 | |

State $S_0$

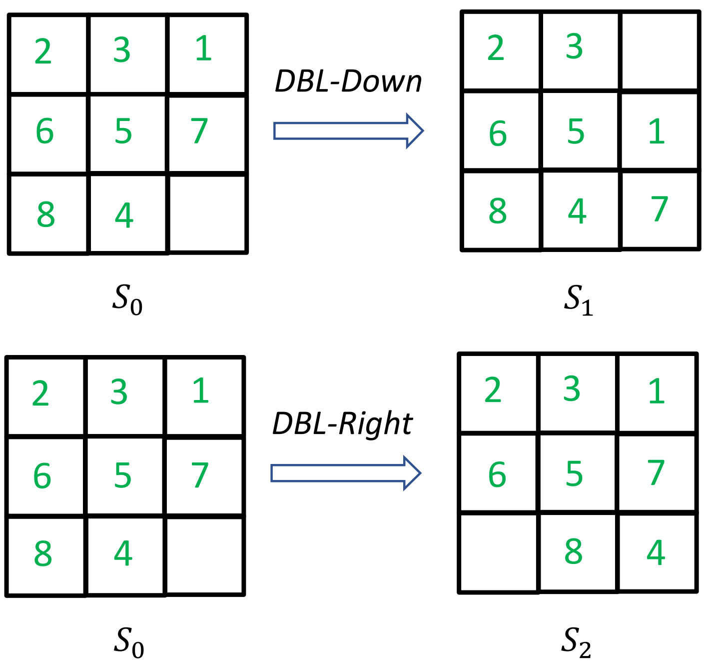
$h_1(S_0) = 8$

$h_2(S_0) =$ $\underline{1\ 2\ 3\ 4\ 5\ 6\ 7\ 8}$

2 1 1 2 2 2 3 1 = 14

(b) [**6 pts**] The number inversions in the goal state $S_g$ is _____6_____. The number of inversions in $S_0$ is ____7____.

(c) [**10 pts**] Recall the four actions in the 8-puzzle: *Left*, *Right*, *Up*, and *Down*, which slide a neighboring tile into the empty square in different directions. Now we add four double-move actions*: DBL-Left*, *DBL-Right*, *DBL-Up*, and *DBL-Down.* Every new action slides two adjacent tiles simultaneously in the same direction and by one square each, so one of the tiles will occupy the (previously) empty square. The figure below on the left illustrates two transitions from the state $S_0$ to the states $S_1$ and $S_2$, respectively, under the actions *DBL-Down* and *DBL-Right.*




(c1) [**4 pts**] Explain why the heuristic $h_2$ is no longer admissible with the four double-move actions now included.

Because the double moves present the possibility to complete the puzzle in fewer moves, the heuristic may overestimate as it's built around single moves

(c2) [**6 pts**] Make use of $h_2$ to construct a new heuristic function $h_3(S)$ that is admissible for the set of the eight actions.

~~For each tile's manhattan distance~~
~~> 0, subtract 1~~
~~h3(S) = max( h(2) – 8, 0 )~~

h3(S) = h2(S) / 2

4. **[16 pts]** *Alpha-Beta Pruning*

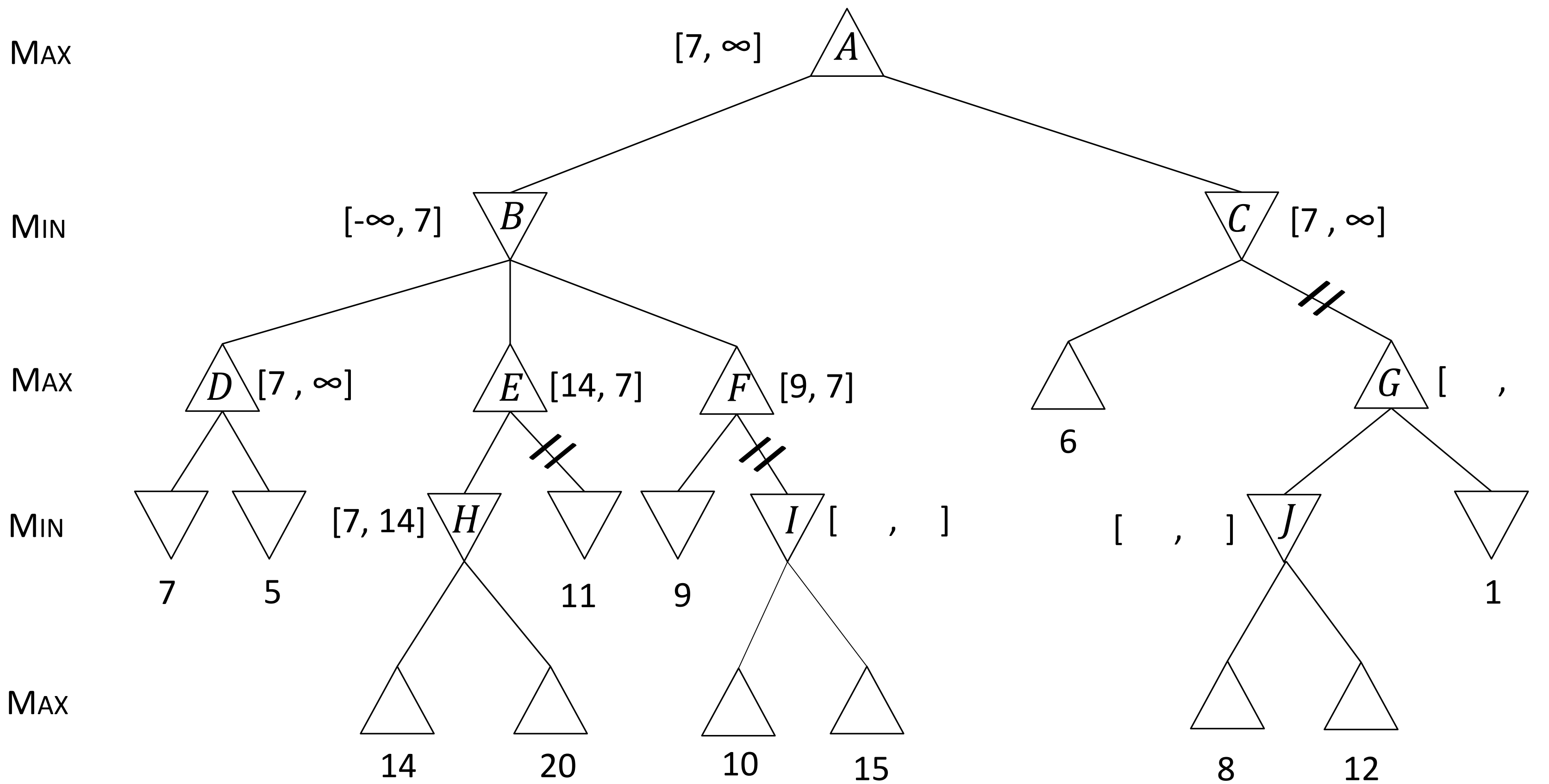
You are given a minimax search tree shown on the next page. The tree has ten internal nodes $A, B, \ldots, J$. Not all terminal states (leaves) are at the same depth. Execute the alpha-beta pruning algorithm.

(a) [**6 pts**] Mark all the subtrees (including leaves) that have been pruned. You may, for instance, simply put double slashes \\ or // across the edge entering the root of such a subtree from the above.

(b) [**7 pts**] At every internal node storing a state on which a call MAX-VALUE or MIN-VALUE is invoked, fill inside the bracket [ , ] next to the node the values of $\alpha$ and $\beta$ at the completion of this call.

(c) [**3 pts**] What is the final value for MAX at the root?

**Final value = 7**


MAX
MIN
MAX
MIN
MAX
[7, ∞]
A
[-∞, 7]
B
C
[7 , ∞]
D
[7 , ∞]
E
[14, 7]
F
[9, 7]
G
[ , ]
6
[7, 14]
H
I
[ , ]
[ , ]
J
7
5
11
9
1
14
20
10
15
8
12

5. **[16 pts]** *Propositional Logic*

Consider the knowledge base (KB) below:

1. The humidity is high or the sky is cloudy.
2. If the sky is cloudy, then it will rain.
3. If the humidity is high, then it is hot.
4. It is not hot.

(a) [**4 pts**] Convert the above four statements in the KB into propositional sentences, using the atomic sentences with their meanings defined in the lower-left table. Write the converted propositional sentences below to the right next to the sentence numbers.

| | |
|---|---|
| Humidity | "The humidity is high." |
| Cloudy | "The sky is cloudy." |
| Rain | "It will rain." |
| Hot | "It is hot." |

1. Humidity V Cloudy
2. Cloudy => Rain
3. Humidity => Hot
4. ¬Hot

(b) [**4 pts**] Convert each of the proportional sentences into conjunctive normal form. In case one sentence generates multiple clauses, write out each clause separately.

1. Humidity ∨ Cloudy
2. ¬Cloudy ∨ Rain
3. ¬Humidity ∨ Hot
4. ¬Hot

⋮

(c) [**8 pts**] Prove `Rain`, i.e., "`It will rain`", using resolution (by refutation). Do this by constructing a resolution tree to the right or on the next page that ends with an empty clause (∅).