**EXAM 1** CSE 5290 (AI) Spring 2018 [6 questions, 60 points]

Take home, Feb 13 Tuesday – 17 Saturday midnight

WRITE YOUR NAME and LAST 4 DIGITS OF ID:

*Submission: One single file to be uploaded on canvas by the due time. Any diagram or picture should be scanned-pasted within the same submitted file.*

**Q1.** Maze path finding *(i)* with *DFS*, and *(ii)* with *A\** search using *Manhattan* distance as heuristics. [Example, start to goal Manhattan distance below is 18.] [10]

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  | 1(goal) |
|  |  |  |  | 1 | 1 |  |  | 1 |  |
|  |  |  | 1 |  |  |  |  | 1 | 1 |
|  |  | 1 |  | 1 | 1 | 1 | 1 | 1 |  |
|  |  | 1 |  |  |  |  |  |  |  |
|  |  | 1 | 1 | 1 |  | 1 | 1 | 1 |  |
|  |  |  |  | 1 | 1 |  |  |  |  |
|  |  |  |  | 1 |  |  |  | 1 |  |
|  | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |  |
| 1(start) |  |  |  |  |  |  |  |  |  |

 **y**

 ***Coordinate system***

 **Start: (0,0) is the left-bottom corner cell Goal: (9,9) right-up corner cell**

 **x**

**1 indicates allowed cell to go to.**

**Agent may move by one cell at a time by up-down-left-right-diagonal on only cells with 1s. Note again, diagonal movement is allowed here.**

**When in conflict, the order of preference for visiting a child should be counter-clockwise starting with the cell on right (++x, same y).**

***Input: Above maze.***

***Output*: Two search trees, with visited coordinates as their nodes, according to the order of visits by the two respective algorithms.**

**You are welcome to use codes and solve this problem with them.**

**Q2.** If f(s), g(s) and h(s) are *admissible* heuristics, then which of the following are also guaranteed to be admissible heuristics: [10]

a. f(s) + g(s) + h(s)

b. f(s)/6 + g(s)/3 + h(s)/2

c. max(f(s), g(s), h(s))

d. min(f(s), g(s), h(s))

e. f(s)/3 + g(s)/3 + h(s)/3

f. f(s)\*g(s)\*h(s)

g. min(f(s), g(s) + h(s))

h. max(f(s), g(s) + h(s))

i. min(f(s), 2g(s) + h(s))

j. max(f(s), g(s) + 2h(s))

**Q3.**  You have three jugs *A, B and C*, measuring 12, 8 and 3 liters respectively, and a water faucet. You may fill any jugs, or empty one to another or to the ground. You need to measure out exactly one liter by only these operations starting with all empty jugs. Use a search space over (x, y, z) where these three numbers *x, y and z* indicate contents of the three jugs A, B and C respectively.

All three jugs are empty in the beginning.

What is your start state and goal state(s)? (Goal states may be more than one.)

Draw a search tree to reach a goal state from the start state. No specific algorithm is suggested here.

Note, the search process is not online, or a committed state is not unrecoverable. In other words, the algorithm may backtrack or retract its decision. [10]

**Q4.** Consider the *constraint reasoning* problem of *4* rooks not attacking each other on a *4 x 4* chessboard (rooks attack along straight lines on chessboard – horizontally and vertically, but only up to a visible piece). Considering each column as a variable (v1-v4, as shown below) and row numbers as their domains {1,2,3,4, bottom to top}.

Draw the corresponding input *constraint graph* showing domain values of nodes, and constraints (as *allowed* pairs of values) on arcs.

Run arc-consistency algorithm showing the resulting *arc-consistent network*. [10]

 v1 v2 v3 v4

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Q5.** Run a *path consistency* algorithm and draw the resulting *path consistent network* for the input below. Dashed arcs are *unconstrained* in the input, meaning the constraints on them are the same {(r,r),(r,b),(b,b),(b,r)}, every possibility being allowed. [10]

B {r,b}

{(r,b), (b,r)}

{(r,r), (b,b)}

D {r,b}

A {r,b}

{(r,b), (b,r)}

{(r,r), (b,b)}

C {r,b}

**Q6.** Derive the resulting *qualitative temporal constraint* on the arc *~~AC~~* *AD* in the following input *qualitative time-interval network*. Composition table will be needed unless you can derive the required atomic compositions intuitively.

Do not run the full path-consistency algorithm. [10]

B

o

o|b

D

A

oi

mi | b

C