CSC 520 Homework Assignment 2

Due: September 26, 2017
Unity ID: rsdates

1. **[50 points]**

   1. **[10 points]** Experiment with executing your implementation of A* to find various paths, until you understand the meaning of the output. Are there any pairs of cities (A,B) for which the algorithm finds a different path from B to A than from A to B? Are there any pairs of cities (A,B) for which the algorithm expands a different total number of nodes from B to A than from A to B?

      Answer

      When exploring the A* algorithm I wasn't able to find any cities in which the path from A to B was different than the path from B to A. This may be because the algorithm is optimal. However, the total number of nodes expanded vary on the path from Orlando to japan and vice versa. For the path Japan to Orlando, expand count is 133 which is smaller than the opposite path where expand count is 1241. I am not surprised that the paths are the same, due to the high success of this particular algorithm. I didn't not expand the expand count to vary so different from city to city. I guess this would be the case for more populated areas that A* must take into account.

   2. **[10 points]**Change your code so as to implement greedy search, as discussed in the web notes.

      Answer

      See Code

   3. **[10 points]** Do enough exploration to find at least one path that is longer using greedy search than that found using A*, or to satisfy yourself that there are no such paths. Find at least one path that is found by expanding more nodes than the comparable path using A*, or satisfy yourself that there are no such paths. If there is such a path, list the nodes in the path and the total distance.

      Answer

      I was not able to find a definite path that the greedy algorithm was longer than A*. I also could not find any path in which the expand counts was higher for greedy than A*.However, I did find a path that greedy runs into an infinite loop. (Japan to Omaha) This could be considered a longer path, but it doesn't actually reach the goal node. The greedy algorithm would typically return shorter paths, which makes sense when the start and goal are fairly close and there is better heuristic estimates. A* however will perform a thorough search to ensure its the best path, so it will in effect expand more nodes before returning the best path.

   4. **[10 points]**Change your code so as to implement uniform cost search, as discussed in the web notes.

      Answer

      See code.

   5. **[10 points]**Do enough exploration to find at least one path that is longer using uniform cost than that found using A*, or to satisfy yourself that there are no such paths. Find at least one path that is found by expanding more nodes than the comparable path using A*, or satisfy yourself that there are no such paths. If there is such a path, list the nodes in the path and the total distance.

      Answer

      In my exploration I could not find any paths that had a different solution path, however I did find a path that had more expansion of nodes on uniform cost search algorithm than A*. The path from Ottawa to thunder bay expanded 28 times which is greater than the A* of 13. The uniform cost algorithm performed well for this map because of the amount of access roads into a node. I feel that uniform cost performed well because there are a lot of long stretch cities making it easy for the algorithm to find paths.

      Uniform

      Expanded Nodes: ottawa, montreal, toronto, albanyNY, buffalo, rochester, boston, cleveland, providence, albanyNY, newHaven, saultSteMarie, columbus, pittsburgh, boston, stamford, dayton, providence, montreal, cincinnati, newHaven, ottawa, indianapolis, stamford, philadelphia, newYork, baltimore, thunderBay

      Number of nodes expanded: 28

      Solution Path: ottawa, toronto, saultSteMarie, thunderBay

Solution Path Node Count: 4
Distance from source to destination: 1147
A*
Expanded Nodes: ottawa, montreal, toronto, buffalo, albanyNY, rochester, saultSteMarie, boston, albanyNY, montreal, cleveland, providence, thunderBay
Number of nodes expanded: 13
Solution Path: ottawa, toronto, saultSteMarie, thunderBay
Solution Path Node Count: 4
Distance from source to destination: 1147

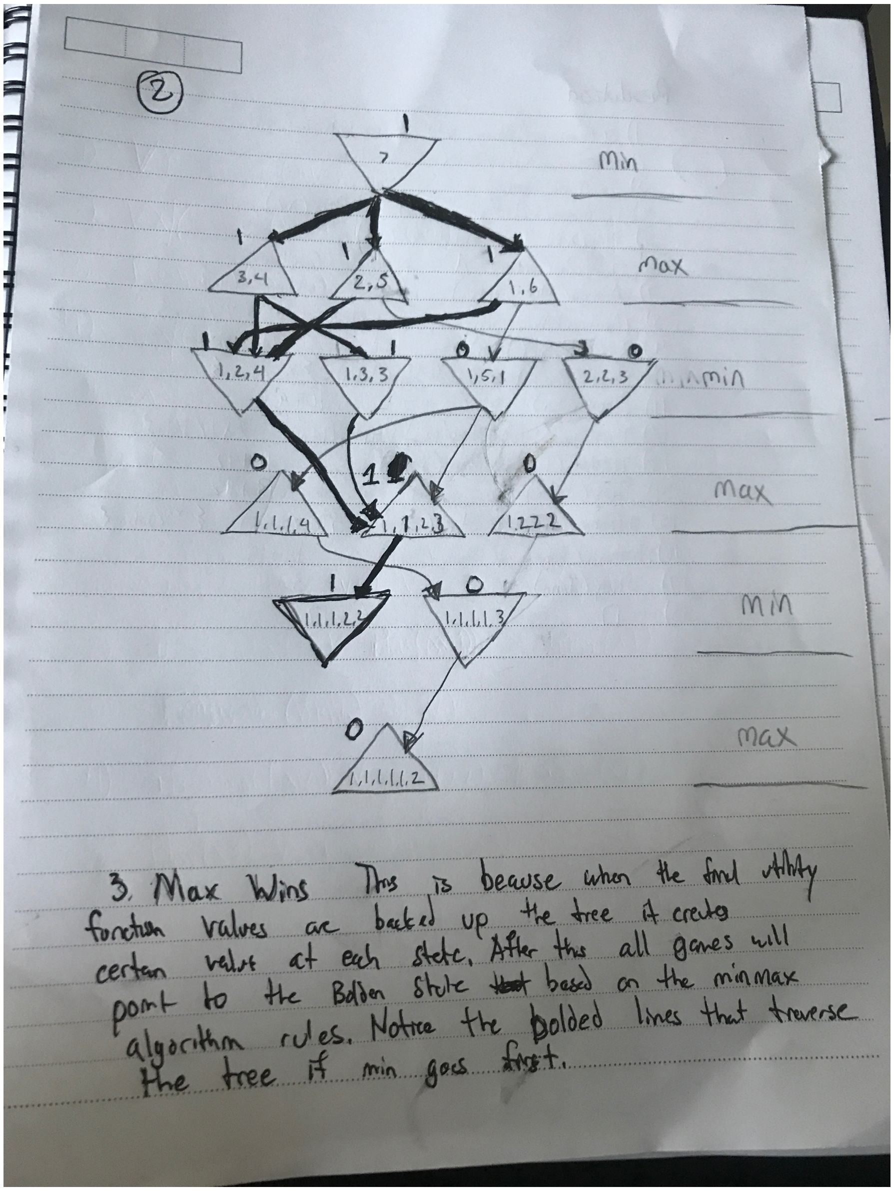
2. **[15 points]** Nim is a two-player game and the rules are as follows: The game starts with a single stack of 7 tokens. At each move a player selects one stack and divides it into two non-empty, non-equal stacks. A player who is unable to move loses the game.

   (a) Draw the complete search tree for Nim.

   (b) Assume two players, Min and Max, play Nim and Min performs the first move. If a terminal state in the search tree developed above is a win for Min, a utility function of zero is assigned to that state. A utility function of 1 is assigned to a state if Max wins the game. Apply the min-max algorithm to the search tree to assign utility functions to all states in the search tree.

   (c) If both Min and Max play a perfect game (optimally), who will win? Explain your answer.

   Answer



3. **[10 points]**] Convert the following set of sentences to Conjunctive Normal Form and use resolution to prove the conclusion: $\neg R \vee \neg S$

   Try to find a shorter proof than the 4-Part Heuristic will give you.

   $T \vee \neg Y \rightarrow \neg S$

   $R \leftrightarrow (S \vee X)$

   $X \rightarrow U$

   $T \wedge Y \rightarrow \neg S$

   $X \rightarrow S$

   Answer

(3) $\neg 1$

1. $\neg T \vee \neg Y \rightarrow \neg S$
2. $R \leftrightarrow (S \vee X)$
3. $X \rightarrow U$
4. $T \wedge Y \rightarrow \neg S$
5. $X \rightarrow S$

~~CNF~~

1) $T \wedge Y \vee \neg S$
2) $\neg R \vee S \vee X$
3) $\neg S \wedge \neg X \vee R$
4) $\neg X \vee U$

$\neg T \vee \neg Y \vee \neg S$

$\neg X \vee S$

CNF (1-8)

1) $T \vee \neg S$
2) $Y \vee \neg S$
3) $\neg R \vee S \vee X$
4) $\neg S \vee R$
5) $\neg X \vee R$
6) $\neg X \vee U$
7) $\neg T \vee \neg Y \vee \neg S$
8) $\neg X \vee S$

Negate Conclusion

$\neg R \wedge \neg S$

$R \vee S$

9) $R \vee S$ Negate Conclusion
10) R 9+4
11) $S \vee X$ 10+3
12) S 11+8
13) T 12+1
14) Y 12+2
15) $\neg Y \vee \neg S$ 13+7
16) $\neg S$ 15+14
17) $\square$ 16+12

4. **[10 points]** Use Propositional Logic to determine whether or not the following set of requirements is logically consistent. In other words, represent the following sentences in Propositional Logic, convert to Conjunctive Normal Form, and run Resolution until a contradiction is derived, or else show one model of all the expressions showing that no contradiction exists.

(4)

a.

1. $a \to p$
2. $e \to w$
3. $m \to w \vee a$
4. $\neg e \to m$
5. $\neg w$
6. $\neg p$

4 part

CNF

1) $\neg a \vee p$
2) $\neg e \vee w$
3) $\neg m \vee w \vee a$
4) $e \vee m$
5) $\neg w$
6) $\neg p$

---

7) $\neg a$ 6 + 1
8) $\neg m \vee w$ 7 + 3
9) $\neg m$ 8 + 5
10) e 9 + 4
11) w 10 + 2
12) $\square$ 11 + 5

Contradiction exists

5. **[15 points]** Write the following English sentences in First Order Logic, convert those First Order Logic sentences into CNF, and prove the conclusion by resolution and the 4-part heuristic presented in class. Number your clauses, and indicate explicitly step-by-step what resolves together, under what substitution.

Answer

5

5a

1) $\forall Y \ \text{Admires}(\text{Mary}, Y) \rightarrow \text{BaseballStar}(Y)$

2) $\forall X [(\text{Freshman}(X) \land \neg \text{Pass}(X)) \rightarrow \neg \text{Play}(X)]$

3) $\text{Freshman}(\text{John})$

4) $\forall X [(\text{Freshman}(X) \land \neg \text{Study}(X)) \rightarrow \neg \text{Pass}(X)]$

5) $\forall X \ \neg \text{Play}(X) \rightarrow \neg \text{BaseballStar}(X)$

Conclusion: $\neg \text{Study}(\text{John}) \rightarrow \neg \text{Admires}(\text{Mary}, \text{John})$

CNF

1) $\neg \text{Admires}(\text{Mary}, y1) \lor \text{BaseballStar}(y1)$

2) $\neg \text{Freshman}(x1) \lor \text{Pass}(x1) \lor \neg \text{Play}(x1)$

3) $\text{Freshman}(\text{John})$

4) $\neg \text{Freshman}(x2) \lor \text{Study}(x2) \lor \neg \text{Pass}(x2)$

5) $\text{Play}(x3) \lor \neg \text{BaseballStar}(x3)$

Conclusion: $\text{Study}(\text{John}) \lor \neg \text{Admires}(\text{Mary}, \text{John})$

5b Resolution

1) ¬Admires(Mary, y1) V BaseballStar(y1)
2) ¬Freshman(x1) V Pass(x1) V ¬Play(x1)
3) Freshman(John)
4) ¬Freshman(x2) V Study(x2) V ¬Pass(x2)
5) Play(x3) V ¬BaseballStar(x3)
6) ¬Study(John) ∧ negate Conclusion
7) Admires(Mary, John) negate Conclusion 6+7
8) ¬Admires(Mary, John) V BaseballStar(John) {y1/John}
9) BaseballStar(John) 7+8
10) Play(John) V ¬BaseballStar(John) {x3/John}
11) Play(John) 10+9
12) ¬Freshman(John) V Pass(John) V ¬Play(John) {x1/John}
13) ¬Freshman(John) V Pass(John) 11+12
14) Pass(John) 13+12
15) ¬Freshman(John) V Study(John) V ¬Pass(John) {x2/John}
16) ¬Freshman(John) V Study(John) 14+15
17) Study(John) 3+16
18) □ 6+17