Indian Institute of Technology Jodhpur
B.Tech.(CSE) 3rd Year, II Semester 2015-16
CS323: Artificial Intelligence Assignment # 1

1. Given a full 5-litre jug with water and an empty 2-litre jug. The jugs have no marking or level indicator. The goal is to fill 2-litre jug with exactly one litre of water. Give the corresponding state diagram. Assume that following are some example of moves: $p_{5,2}$ (pour 5 litre into 2 litre jug), e_2 (empty two litre jug), etc.

2. Given the blocks world indicated in figure 1, it is required to get to goal state from the initial start state. Construct the search tree for:

   (a) BFS

   (b) DFS

   Assume that following rules for moves will be followed by the robot arm for carrying out this job:

   - $stack(x, y)$: stack block x on block y,
   - $lift(x)$: lift-up the block x,
   - $putg(x)$: put block x on ground,
   - $unstack(x, y)$: unstack block x from block y.

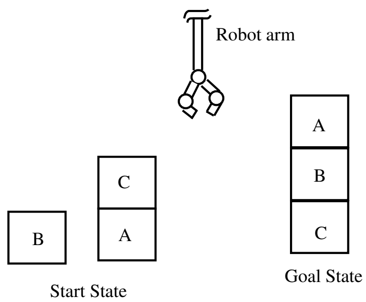

Figure 1: Blocks World.

3. For the graphs shown in figure 2(a) to (d), with start node S and goal node G, find out the average node branching factor and average edge branching factor.

4. Solve the *Towers of Hanoi* problem using DFS, and find out the space and time complexity for this for n disks. Also answer the following:

   (a) What is maximum branching factor for n disks?

   (b) What is average branching factor for $n = 4$ disk?

5. Suggest an architecture of hash-table for storage of nodes found in graph search, and quick searching of the same when it is found again next-time in graph search.

6. Answer the following short review questions.

   (a) In what condition the best-first search becomes the breadth-first?

   (b) What you can infer from the condition: $f(n) = g(n)$?

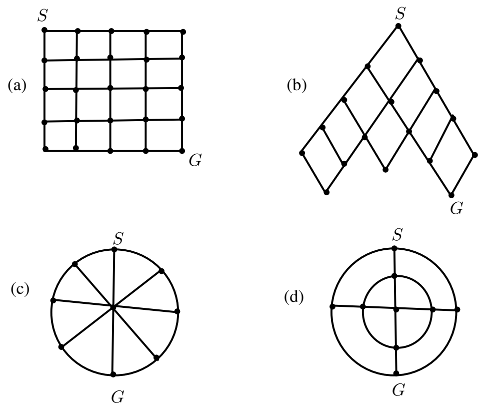

Figure 2: A graph with start node S goal G.

(c) What you can infer from the condition: $f(n) = h(n)$?

(d) In what situation the A^* search become best-first search?

(e) What is the primary drawback of best-first search?

(f) Which search method uses the priority-queue data structure?

7. Suggest a heuristic function for the 8-puzzle that sometimes overestimates, and show how it can lead to a suboptimal solution on a particular case.

8. Give the name of the algorithms that results from each of the following special cases:

   (a) Local beam search with $k = 1$.

   (b) Local beam search with $k = \infty$.

   (c) Simulated annealing with $T = 0$ at all times.

   (d) Genetic algorithm with population size $N = 1$.

9. Explain, how will you use best-first search in each of the following cases? Give the data structure and explain logic.

   (a) PCB design

   (b) Routing Internet traffic

   (c) Scene analysis

   (d) Mechanical theorem proving

10. What type of data structure is suitable for implementing best-first search, such that each node in the frontier is directly accessible, and all the vertices behind it remain in the oder they have been visited.

11. Is there a danger of Local maximum in GA? How does the algorithm tries to avoid it?

12. If there is no solution, will A^* explore the whole graph? Justify.

13. For the graph shown in figure 3, find out whether the A^* search for this graph is,

    (a) Optimal?

    (b) Oder-preserving?

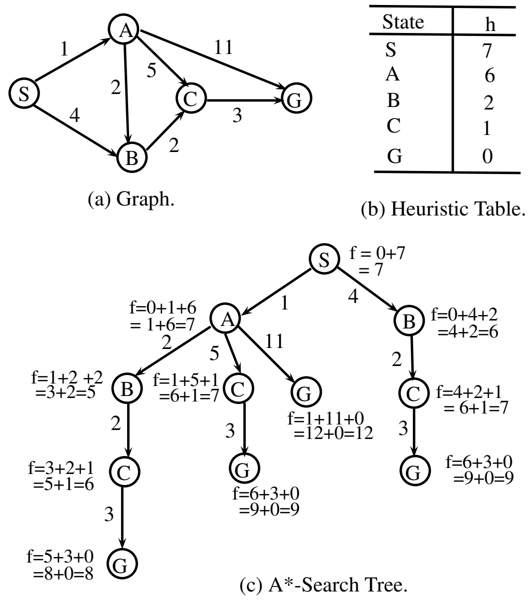

Figure 3: Graph and A*-Search-tree.

(c) Complete?

(d) Sound?

14. Apply the BFS algorithm for robot path planing in the presence of obstacles for a square matrix of 8×8 given in figure 4. Write an algorithm to generate the frontier paths. Assume that each move of robot in horizontal (H) and vertical (V) covers a unit distance, and the robot can take only the H and V moves. The start and goal nodes are marked as S and G. Shaded tiles indicate obstacles, i.e., robot cannot pass through these.

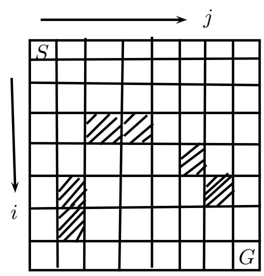

Figure 4: 8×8 tiles, with obstacles in shades.

15. What are the consequences of following special cases of GA-based search?

(a) Only the selection operation is performed in each iteration, based on the fitness value.

(b) Only the cross-over operation is performed in each iteration at a random position.

(c) Only the mutation operation is performed in each iteration at a random bit position.

16. *Sudoku* can be viewed as a binary constraint satisfaction problem.

(a) What are the variables of this CSP?

(b) What are their domains?

(c) How would you translate the requirement that no two of the same digit may occur in the same row, column, or block into binary constraints?

(d) Does the requirement that each digit occur at least once in each row, column, or block have to be directly specified? Why or why not?

17. For what size of n, the n-queen problem has no solution, with n in the range 1-10.

18. The figure 5 shows 9-tiles in the form of variables $(X_1 \dots X_9)$ for CSP. Write the steps of an algorithm to color these tiles using R, G, B colors such that no adjacent tiles have same color. Also, so the final assignment of colors to these tiles. (Note: A tile is adjacent to other if it is in the same row or column and near to the other.)

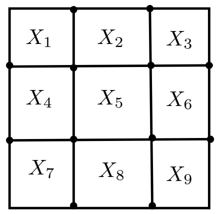

Figure 5: Tiles to be colored.

Note: Submission deadline March 28-03-2016, 11.59PM online. Those who have last digit odd in their roll number will submit the solution of odd numbered problems, and those having even last digit will submit the even numbered problem's solutions.