**CSE112 Artificial Intelligence**， **Week 5 2019**

Exercises and Tutorial Questions

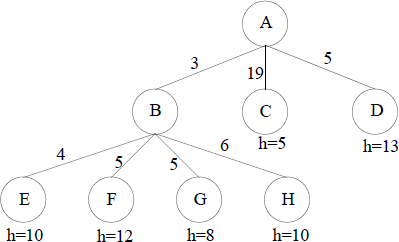
1. **Review questions**
   1. For a general search problem, state which of iterative deepening (ID) or depth-first search (DFS) is preferred under which of the following conditions:
      1. A shallow solution is preferred.
      2. The search tree may contain large or infinite branches.
   2. Say we define the evaluation function for a heuristic search problem as

*f* (*n*) = (1 − *w*)*g*(*n*) + *w h*(*n*)

where *g*(*n*) is the cost of the best path found from the start state to state *n*, *h*(*n*) is an admissible heuristic function that estimates the cost of a path from *n* to a goal state, and 0. 0 ≤ *w* ≤ 1. 0. What search algorithm do you get when

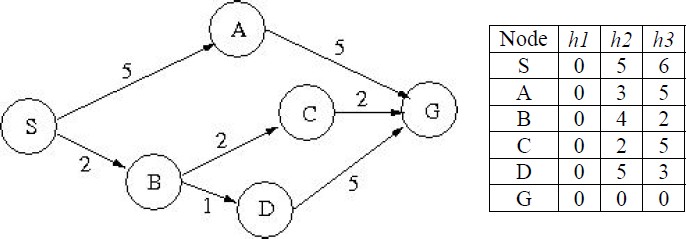
(i) *w* = 0. 0

1. *w* = 0. 5
2. *w* = 1. 0
   1. Consider the following search tree, where each arc is labelled with the cost of the corresponding operator, and the leaves are labelled with the value of a heuristic function, *h*. For uninformed searches, assume children are expanded left to right. In case of ties, expand in alphabetical order.



Which node will be expanded next by eachfolltohweing search methods?

1. Depth-First Search
2. Greedy Best-First Search
3. Uniform-Cost Search
4. A\* Search
   1. Consider the following search space where we want to find a path from the start state S to the goal state G. The table shows three different heuristic functions *h1, h2*, and *h3*.



1. What solution path is found by Greedy Best-first search using *h2*? Break ties alphabetically.
2. What solution path is found by Uniform-Cost search? Break ties alphabetically.
3. Give the three solution paths found by algorithm A\* using each of the three heuristic functions, respectively. Break ties alphabetically.
   1. Question on A\* Heuristics

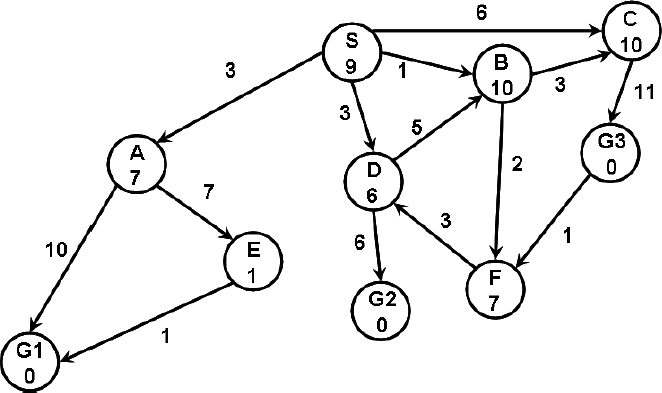
In standard A\* search the objective function at each node *n* is *f*(*n*) = *g*(*n*) + *h*(*n*), where *g*(*n*) is the cost from start to this node, and *h*(*n*) is an **admissible** heuristic estimating the cost from *n* to a goal. Now let us use a different objective function:

*f*(*n*) = *w  g*(*n*) + (100 − *w*)  *h*(*n*) where 0 ≤ *w* ≤ 100.

1. What search algorithm do you get when *w* = 0?
2. What about when *w* = 50?
3. What about when *w* = 100?
   1. Consider the search graph below, where S is the start node and G1, G2, and G3 are

goal states. Arcs are labelled with the cost of traversing them and the heuristic cost to a goal is shown inside the nodes.

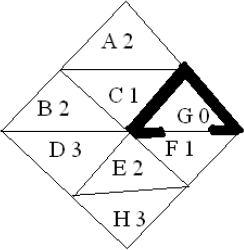
For each of the three search strategies below, indicate which of the goal states is reached:



1. Breadth-first search. Goal reached:
2. Uniform cost search. Goal reached:
3. A\* search. Goal reached:
   1. Consider the following path-finding problem. One can move from one small triangle to another if they share a vertex (e.g., *A* can go to *B* and *C*). However, the goal *G* can only be accessed from *F*. The number after the letter is the heuristic function value for that state. The actual cost of each move is as follows:

A move down one level (e.g. *A* →*C or B →E*) costs 1

A move sideways on the same level (e.g. *C* → *B* or *E* → *F*) costs 2 A move up one level (e.g. *B* → *A* or *C* → *A* ) costs 3



(a). Perform **Depth-First Search**, starting from *A*, using path-checking to avoid repeated states if they occur on the path back to the root in the search tree. Expand successors in alphabetical order. Show your search tree, and *circle* states that are expanded. What is the *cost* of your solution path?

(b) Perform **A\* Search**, starting from *A*. Break ties alphabetically. Show the expanded states and the priority queue contents at each step. What is the cost of your solution path?

# Consider the following questions:

1. Consider the 8-puzzle in which there is a 3 x 3 board with eight tiles numbered 1 through 8. The goal is to move the tiles from a start configuration to a goal configuration, where a move consists of a horizontal or vertical move of a tile into an adjacent position where there is no tile. Each move has cost 1.
2. Is the heuristic function defined by  **admissible**, where *di* is the number of vertical plus the number of horizontal moves of tile *i* from its

current position to its goal position assuming there are no other tiles on the board, and 0 ≤ *i* ≤ 1 is a constant weight associated with tile *i*? Explain briefly why or why not.

1. Is the heuristic defined by *h*(*n*) = 8 – cost(*n*) **admissible**, where cost(*n*) is the cost from start to node *n*? Explain briefly why or why not.
2. Given two arbitrary admissible heuristics, *h1* and *h2*, which composite heuristic is better to use, max(*h1*, *h2*), (*h1* + *h2*)/2, or min(*h1*, *h2*)? Explain briefly why.

# Tutorial Questions on Knowledge Representation Q1. What is knowledge?

**Q2. What types of knowledge can be categorized?**

**Q3. What kind of different levels of knowledge we need to discuss in AI?**

**Q4. Why natural language has not been chosen to represent knowledge in the history of AI development?**

**Q5. What are the desirable characteristics of an appropriate KR scheme?**

**Q6. Explain the important components of Rule-Based System**

**Q7. What Is an Expert System?**

**Q8. Discuss the differences between general software systems and Expert systems.**

**Q9. Describe stages of the knowledge engineering process.**

**Q10. Open question: In your opinion, what is role of knowledge representation in AI?**

**Q11. Open question: Make a general investigation about Knowledge Graph (** 知 识图谱**) and discuss its relationship with AI applications.**

**Q12. What is the meaning of meta knowledge?**