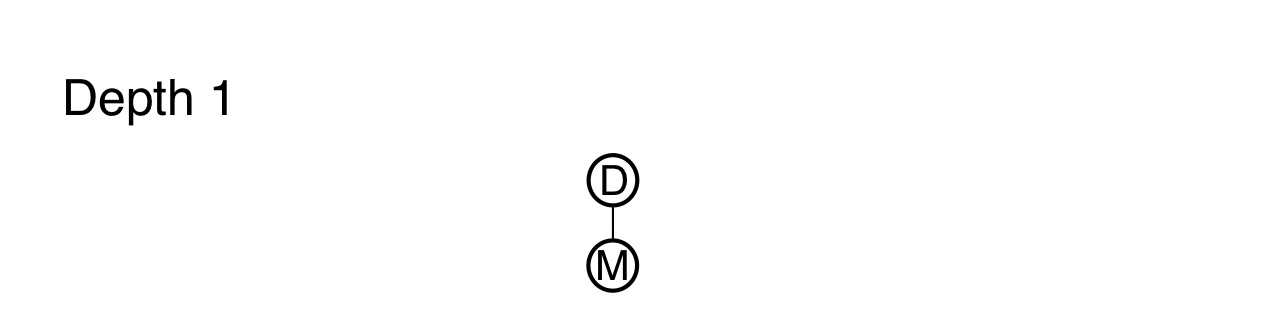
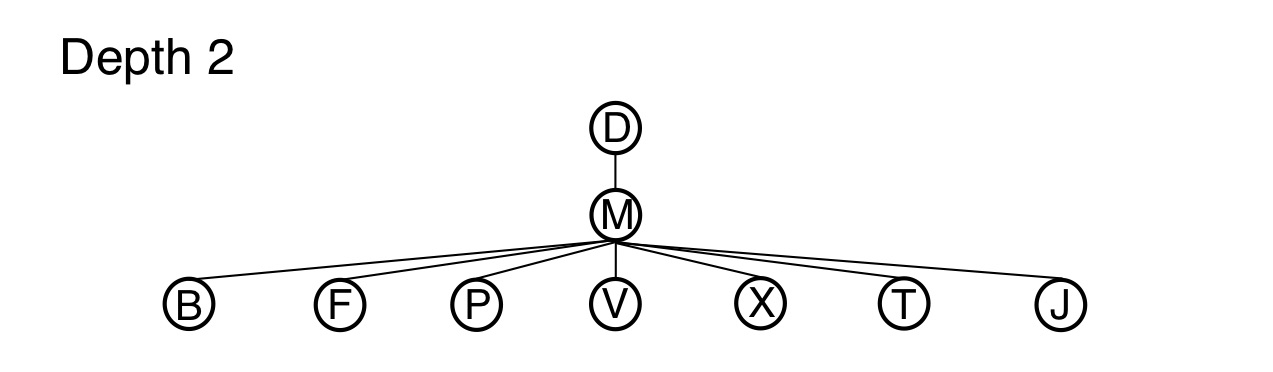
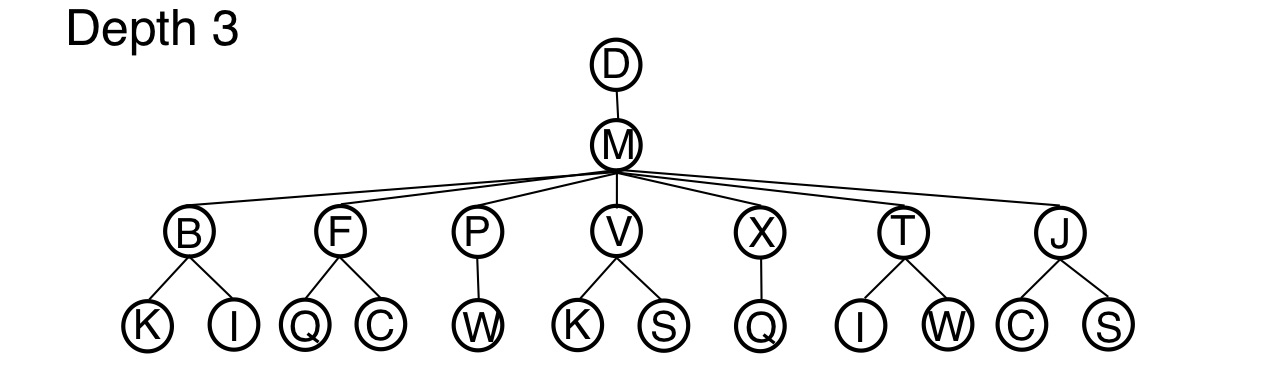
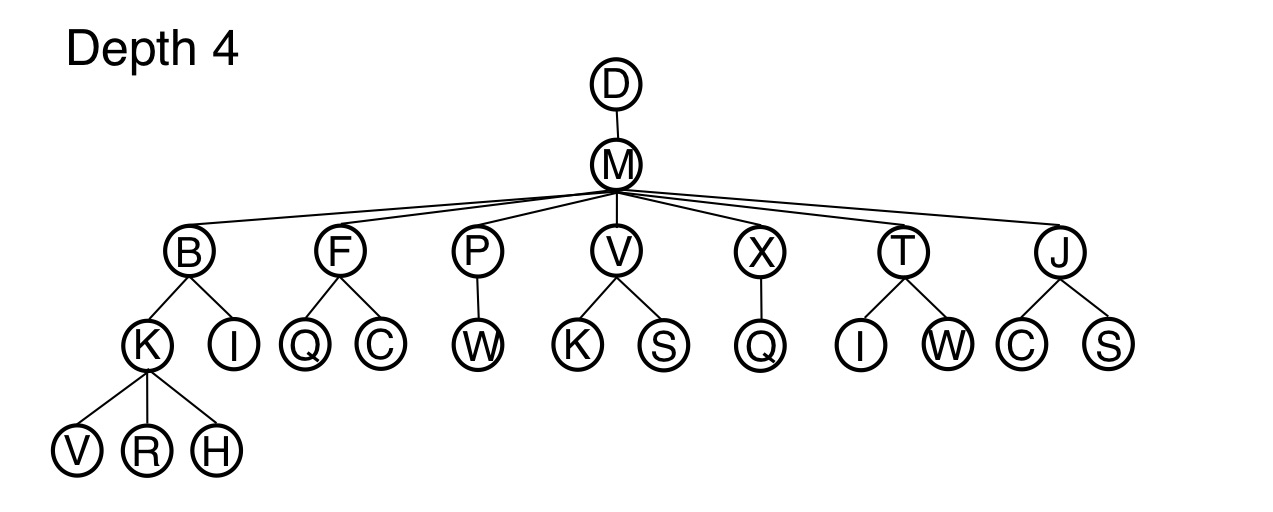
Problem 1:

1. Expanded: D, M, B, K, V, S, H, A, L, U, R.

Solution Found: D, M, B, K, V, S, H, A, L, U, R.

1. Solution is found at depth 4: D, M, B, K, R
2. The answer is no, h(n) is not admissible. Intuitively, this heuristic function models the idea that, “the close a tile is to the goal, the lower its heuristic value”. And clearly this is not the case because of the way a knight moves. Secondly, the condition of an admissible heuristic function is that h(n) ≤ h\*(n). Given the problem and the suggested heuristic function, there are cases where this condition does not hold. For example, given that tile K has coordinates (0, 2) and tile R has coordinates (2, 3).

h(K) = | 0 – 2| + | 2 – 3|

= 3

h\*(K) = 1 (it takes one move to reach the goal, and each move has a cost of 1)

In this example, h(K) > h\*(K), thus, it is not an admissible heuristic function.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Order of addition in frontier | Cell | Order of expansion | g(n) | h(n) | f(n) = (g(n) + h(n)) |
| 1 | D | 1 | 0 | 4 | 4 |
| 2 | M | 2 | 1 | 1 | 2 |
| 3 | J |  | 2 | 4 | 6 |
| 4 | T | 4 | 2 | 2 | 4 |
| 5 | X | 8 | 2 | 2 | 4 |
| 6 | V | 5 | 2 | 2 | 4 |
| 7 | P | 3 | 2 | 2 | 4 |
| 8 | F | 11 | 2 | 4 | 6 |
| 9 | B | 10 | 2 | 4 | 6 |
| 10 | W | 7 | 3 | 1 | 4 |
| 11 | I | 14 | 3 | 3 | 6 |
| 12 | S | 6 | 3 | 1 | 4 |
| 13 | K |  | 3 | 3 | 6 |
| 14 | L |  | 4 | 2 | 6 |
| 15 | H | 13 | 4 | 2 | 6 |
| 16 | N |  | 4 | 2 | 6 |
| 17 | Q | 9 | 3 | 1 | 4 |
| 18 | C | 12 | 3 | 3 | 6 |
| 19 | E |  | 5 | 5 | 10 |
| 20 | A |  | 5 | 5 | 10 |
| 21 | R | 15 | 4 | 0 | 4 |

Solution Path: D, M, T, I, R.

Problem 2:

1. **Write a detailed formulation for this search problem. (Use formal notation)**

**States:** Each state can be defined as an (x, y) combination, where x and y correspond to cities on the map.

**Actions:** For each turn, simultaneously move x and y to a neighboring city.

**Transition Model:** (x, y/ x has x’ && y has y’) -> (x’, y’), where x’ is a neighboring city of x, and y’ is a neighboring city of y.

**Initial State:** (i, j), such that i and j are the city in which the to friends live in respectively.

**Goal State:** If X and Y are both equal, then the goal state is met

**Path Cost, g(n):** The path cost is the total distance from the starting node to the current node, or in this case the function d(i, n), where i is the starting node, and n is the current node.

1. **Let D(i, j) be the straight-line distance between cities i and j. Which of the following heuristic functions are admissible?**

(i) 1

(iii) D(i, j)/2

1. **Are there completely connected maps for which no solution exists? Briefly explain why or why not?**

Yes, there exist completely connected maps for which no solution exists. An example is a map with two nodes and one link. This is because at each turn, the two friends have to call each other after they have moved. This means if the started at opposite nodes; they would be going back and forth between both cities, never meeting each other.

1. **Are there maps in which all solutions require one friend to visit the same city twice? Briefly explain why or why not.**

No, having a person visit a city twice just means that person is doing redundant work. The optimal solution path would not have a person go through the same city twice.