1. Which of the following local search algorithms are complete? Greedy Hill Climbing, Stochastic Hill Climbing, First-Choice Hill Climbing, Random- Restart Hill Climbing, Local Beam Search, Stochastic Beam Search (no explanations requested) [2]

2. Explain briefly how the local search technique Stochastic Beam Search works. [4]

The following questions are about the following state space graph. Let A be the start state and I the goal state. The edge labels indicate step-cost, the vertex labels contain the node identifier in the form of a letter. The order in which successors are generated is counterclockwise, ending at exactly 9 o’clock. Example: E generates first H, then F, then B, and finally D. When sorting by f-value, nodes with equal f-value are ordered such that the earlier a node is generated, the higher its priority. Nodes already on the open list have higher priority than newly added nodes with equal f-value. Heuristic $h(node)$ is defined as the minimum number of steps from node to the goal state; for example, $h(A) = 4$. You may use the following abbreviations without defining them: NGF = No Goal Found, GF = Goal Found.

3. Give the execution trace for Uniform Cost Graph Search (UCGS). [10]

4. Is UCGS optimal for this problem? Explain your answer! [2]

5. Give the execution trace for Greedy Best First Graph Search (GBeFGS) employing heuristic $h$. [8]

6. Is GBeFGS employing heuristic $h$ optimal for this problem? Explain your answer! [1]

7. Give the execution trace for $A^*$ Graph Search ($A^*$GS) employing heuristic $h$. [10]

8. Is for this problem $h$ admissible? Explain your answer! [3]


10. Is $A^*$GS employing heuristic $h$ optimal for this problem? Explain your answer! [1]

11. Define heuristic $h'(node)$ as the maximum number of steps from node to the goal state without taking the same action twice; for example, $h'(A) = 8$. Now define the composite heuristic $h_c(node)$ as $max\{h(node), h'(node)\}$. Is for this problem $h_c$ consistent? [4]

12. This question is about the following graph:

Assuming a bound of [-10,10] on the state eval values, calculate first the bound on node C before evaluating node $D_1$, then after evaluating node $D_1$, and finally after evaluating both node $D_1$ and $D_2$. Show all your calculations for full points! [5]
13. This question and the next two are about the following adversarial search tree. State evaluation heuristic values for the max player are provided in the form of numbers following the letter labels of the states (e.g., A13 indicates that the heuristic value of state A for the max player is 13). The order in which successors are generated is from left to right. Example: A generates first B and then C. Non-quiescent states are indicated by bold circled states. The history table contains all zeros except for AC which is 1, FN which is 1, and FO which is 2.

Give the execution trace for HTQSABDLM(A,3,1,$-\infty$, $\infty$). (That’s DLM, not IDM!) [18]

14. Indicate for HTQSABDLM(A,3,1,$-\infty$, $\infty$) which nodes, if any, get pruned. [5]

15. What is the Principal Variant (PV) found by HTQSABDLM(A,3,1,$-\infty$, $\infty$)? [3]

16. This question and the next are about the following state space graph. Heuristic $h(n)$ is defined by the values following the node labels in the state space graph. Step cost values follow the action labels. Nodes are expanded counter-clockwise, ending at exactly 9 o’clock; example: A generates first action $p$, then action $r$, and finally action $t$. When multiple actions with equal LRTA*-COST are found, use the one found first.

Give the LRTA* trace terminating either when the goal is found or after the 15th call to LRTA*-COST. [18]

17. What is the competitive ratio based on the final state of your LRTA* trace? Explain your answer and make sure to list the final state you are using. [4]