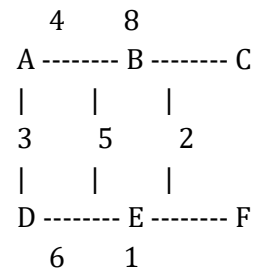
Final Exam Practical Sample Question: Neural Network and A* Search for Shortest Path - Answers

1. Shortest Path Calculation (Manual Heuristic)

Graph Representation:



Heuristic Table:

Node	Heuristic (h_manual)
A	10
B	8
C	5
D	7
E	4
F	0

Steps:

1. Start at Node A: g(A) = 0, h(A) = 10, f(A) = g(A) + h(A) = 0 + 10 = 10.
2. Expand Node A, evaluate neighbors B and D:
   - B: g(B) = 4, h(B) = 8, f(B) = 4 + 8 = 12.
   - D: g(D) = 3, h(D) = 7, f(D) = 3 + 7 = 10.
3. Select Node D (lowest f).
4. Continue to expand Node D and so on, until Node F is reached.

Shortest Path (Manual Heuristic): A → D → E → F

2. Shortest Path Calculation (NN Predicted Heuristic)

Updated Heuristic Table:

Node	Heuristic (h_nn)
A	9.8
B	7.9
C	5.2
D	7.1

E	4.3
F	0.1

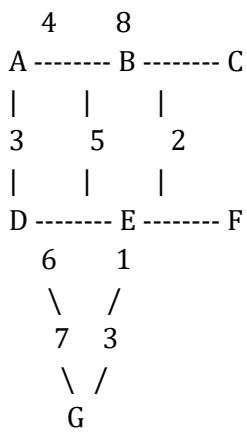
Steps:

1. Start at Node A: g(A) = 0, h(A) = 9.8, f(A) = g(A) + h(A) = 0 + 9.8 = 9.8.
2. Expand Node A, evaluate neighbors B and D:
   - B: g(B) = 4, h(B) = 7.9, f(B) = 4 + 7.9 = 11.9.
   - D: g(D) = 3, h(D) = 7.1, f(D) = 3 + 7.1 = 10.1.
3. Select Node D (lowest f).
4. Continue similarly to reach Node F.

Shortest Path (NN Predicted Heuristic): A → D → E → F

3. Combined Tasks

Updated Graph with Node G:



Steps:

1. Recompute paths considering Node G:
   - g(G) = g(D) + 7 or g(G) = g(F) + 3.
   - Use both manual and NN heuristics to calculate new paths and costs.
2. Shortest Path with Node G:
   - Likely A → D → G → F if this reduces the total cost.

Dynamic Weight Changes:

- If edge weights change (e.g., due to traffic), retrain the neural network using updated datasets that include the new weights and configurations. This allows the NN to adapt and predict heuristics more accurately in real-time.