Report

Question:

Analyze the graph data using these 4 algos, and prepare a detailed report. The report should highlight the pros and cons of applying these different path finding algorithms for about 5 different node pairs taken from the above graph data.

Graph structure:-

- Number of nodes: 45
- Number of edges: 94
- Number of edges in a complete graph: 990
- The graph is connected

Five different node pairs:-

Path	Start Node	Goal Node
1	Delhi	Patna
2	Patna	Delhi
3	Jehanabad	Gaya
4	Gaya	Gaya
5	Calicut	Bundi

1. Delhi to Patna

Algorithms	Time	Memory (in byte)	Path found	Path length	Remarks
BFS	0.0010006	88	yes	4	GBFS is giving minimum time to find the path, but taking much more memory than BFS and DFS. BFS is taking less memory than all and also short path length and DFS is taking more space than all. Here, BFS is optimal.
DFS	0.0010013	256	yes	25	
GBFS	0.0009994	184	yes	16	
A*	0.0010185	120	yes	6	

2. Patna to Delhi

Algorithms	Time(ms)	Memory (in byte)	Path found	Path length	Remarks
BFS	0.0012903	88	yes	4	DFS is taking very less time to find the path. whereas, GBFS takes up a lot of space and time. DFS is also taking up a lot of space. BFS is optimal here as it is giving the shortest path.A* is taking more time than all algorithms to find the path.
DFS	0.0009980	208	yes	19	
GBFS	0.0013122	312	yes	28	
A*	0.0020308	120	yes	6	

3. Jehanabad to Gaya

Algorithms	Time(ms)	Memory (in byte)	Path found	Path length	Remarks
BFS	0.00100183	88	yes	4	DFS is taking less time. BFS is taking less memory than all. GBFS is taking up more space than all. And also its path length is the longest. BFS and A* is giving the optimal path..
DFS	0.00100135	144	yes	11	
GBFS	0.00100779	248	yes	18	
A*	0.00199532	120	yes	4	

4. Gaya to Gaya

Algorithms	Time(ms)	Memory (in byte)	Path found	Path length	Remarks
BFS	0.0009970	64	yes	1	BFS is taking less time than all. All algorithms are consuming equal memory.. Also the path length is the same for these three algo's as the goal node and start node are the same.
DFS	0.0009977	64	yes	1	
GBFS	0.0010023	64	yes	1	
A*	0.0010027	64	yes	1	

5. Calicut to Bundi

Algorithms	Time(ms)	Memory (in byte)	Path found	Path length	Remarks
BFS	0.00099992	72	yes	2	GBFS is taking less time. DFS occupies a large space. BFS and A* is giving an optimal path and also taking less memory. The GBFS path will lead us to a dead end, and no path will be found.
DFS	0.00100135	200	yes	18	
GBFS	0.00099945	56	No	0	
A*	0.00109362	120	yes	2	

Learning from the assignment:

I took help from GPT for implementing the algorithms. I took help from my friends to implement these. I understood the code implementation then I ran the code for different numbers of examples and observed the differences created by all the algorithms. I observed pros and cons for using these algorithms for time, path length and as well as for memory consuming. I observed that on average BFS and A* is giving us the optimal path for traveling one node to another node. GBFS can lead us to a dead end as it goes for the shortest heuristic distance, but can not backtrack when dead end occurs (In the fifth case from Calicut to Bundi). A* is consuming more space than all.