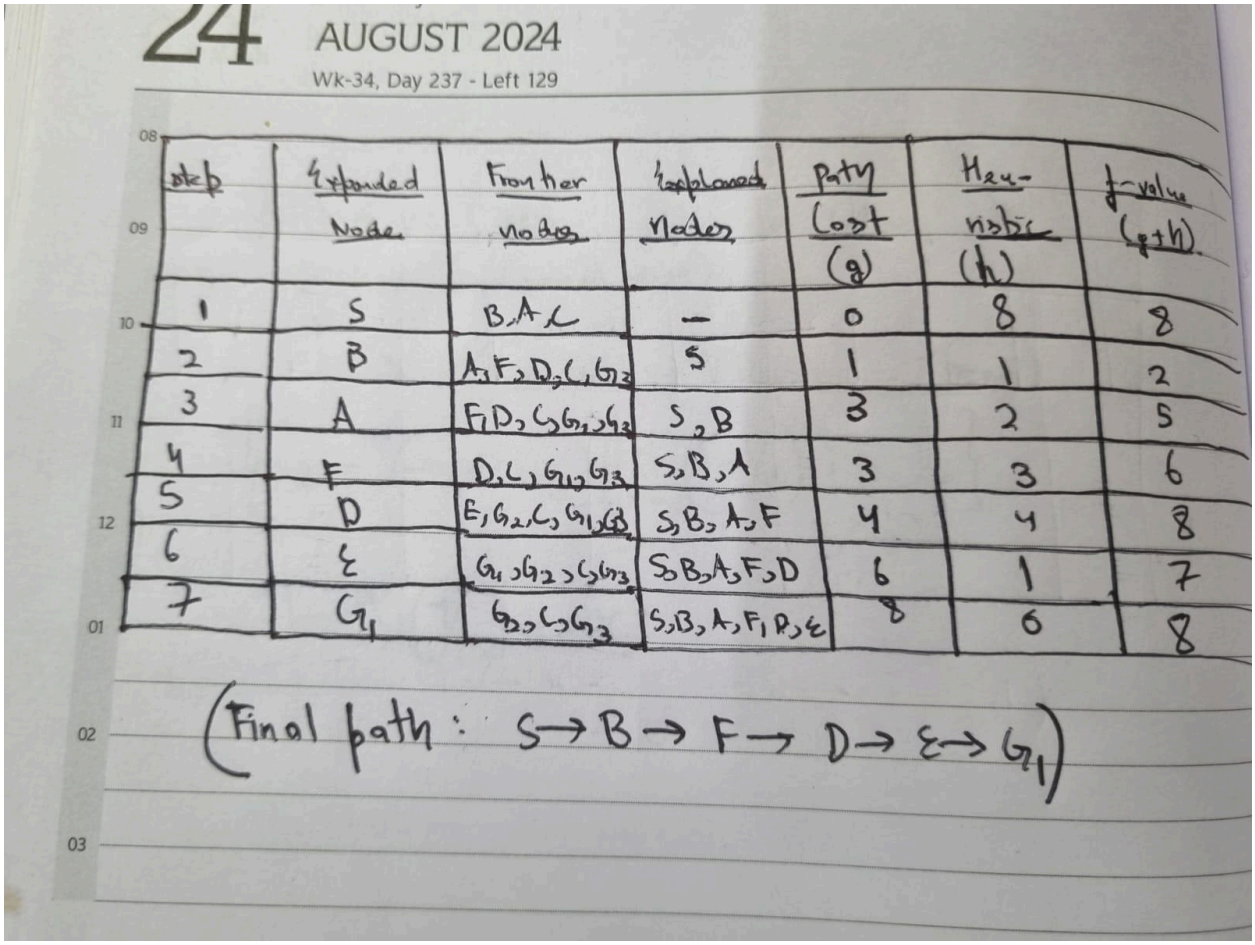
A1 Assignment 1

Keshav Chhabra 2022247

Question1

a) A* search



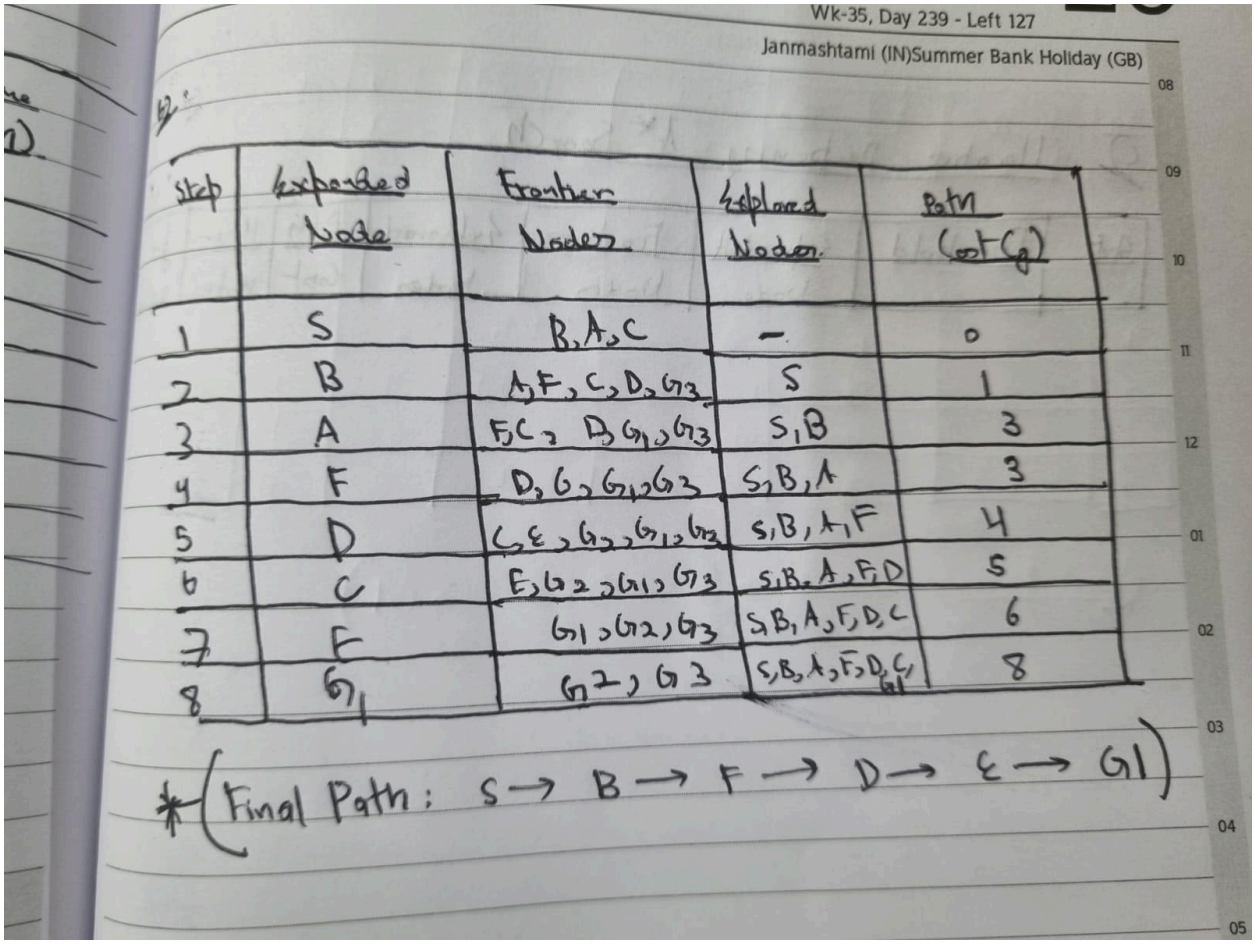
24 AUGUST 2024

Wk-34, Day 237 - Left 129

Step	Expanded Node	Frontier nodes	Explored nodes	Path Cost (g)	Heuristic (h)	f-value (g+h)
1	S	B, A, C	—	0	8	8
2	B	A, F, D, C, G_3	S	1	1	2
3	A	F, D, C, G_2, G_3	S, B	3	2	5
4	F	D, C, G_1, G_3	S, B, A	3	3	6
5	D	E, G_2, C, G_1, G_3	S, B, A, F	4	4	8
6	E	G_1, G_2, C, G_3	S, B, A, F, D	6	1	7
7	G_1	G_2, C, G_3	S, B, A, F, D, E	8	0	8

(Final path : S → B → F → D → E → G_1)

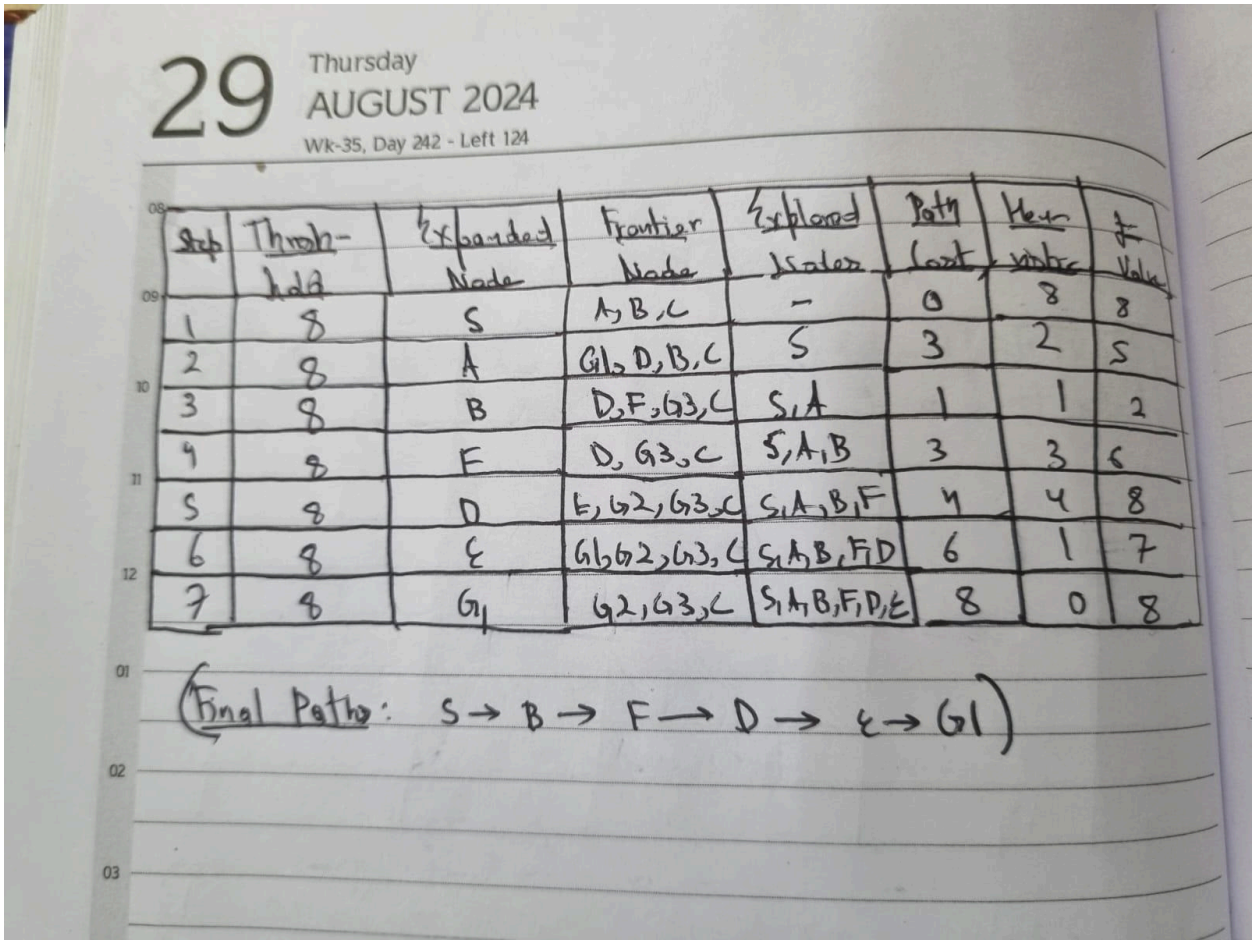
b) Uniform Cost Search



Step	Expanded Node	Frontier Nodes	Explored Nodes	Path cost (g)
1	S	B, A, C	–	0
2	B	A, F, C, D, G3	S	1
3	A	F, C, D, G1, G3	S, B	3
4	F	D, G2, G1, G3	S, B, A	3
5	D	C, E, G2, G1, G3	S, B, A, F	4
6	C	E, G2, G1, G3	S, B, A, F, D	5
7	E	G1, G2, G3	S, B, A, F, D, C	6
8	G1	G2, G3	S, B, A, F, D, C, G1	8

* (Final Path: S → B → F → D → E → G1)

c) Iterative Deepening Search



29 Thursday
AUGUST 2024
Wk-35, Day 242 - Left 124

Step	Threshold	Expanded Node	Frontier Nodes	Explored States	Path Cost	Heuristic	f Value
1	8	S	A, B, C	–	0	8	8
2	8	A	G1, D, B, C	S	3	2	5
3	8	B	D, F, G3, C	S, A	1	1	2
4	8	F	D, G3, C	S, A, B	3	3	6
5	8	D	E, G2, G3, C	S, A, B, F	4	4	8
6	8	E	G1, G2, G3, C	S, A, B, F, D	6	1	7
7	8	G1	G2, G3, C	S, A, B, F, D, E	8	0	8

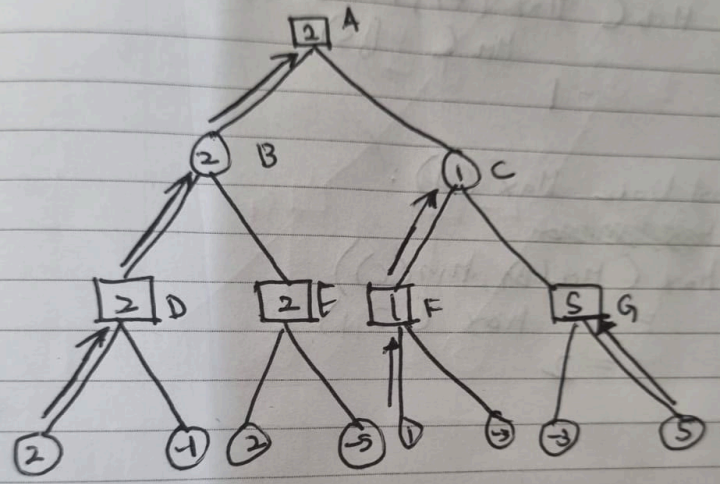
Final Path: S → B → F → D → E → G1

Question 2

Question-2

a) Part A

- MM- Max Algorithm.



Possible moves using MM- Max Algorithm.

- Max Nodes (D, E, F, G)
  - Max (2, -1) → 2
  - Max (2, 5) → 2
  - Max (1, -3) → 1
  - Max (-3, 5) → 5

31 AUGUST 2024

Wk-35, Day 244 - Left 122

- Min Nodes (B, C):
  * Min (Max(D), Max(E))
    = Min (2,2)
    = 2
  * Min (Max (F), Max (G)) =
    Min (6,5)
    = 1

- Root Node Max (A):
  * Max (Min(B), Min(C))
    = Max (2,1)
    = 2

1 Sunday

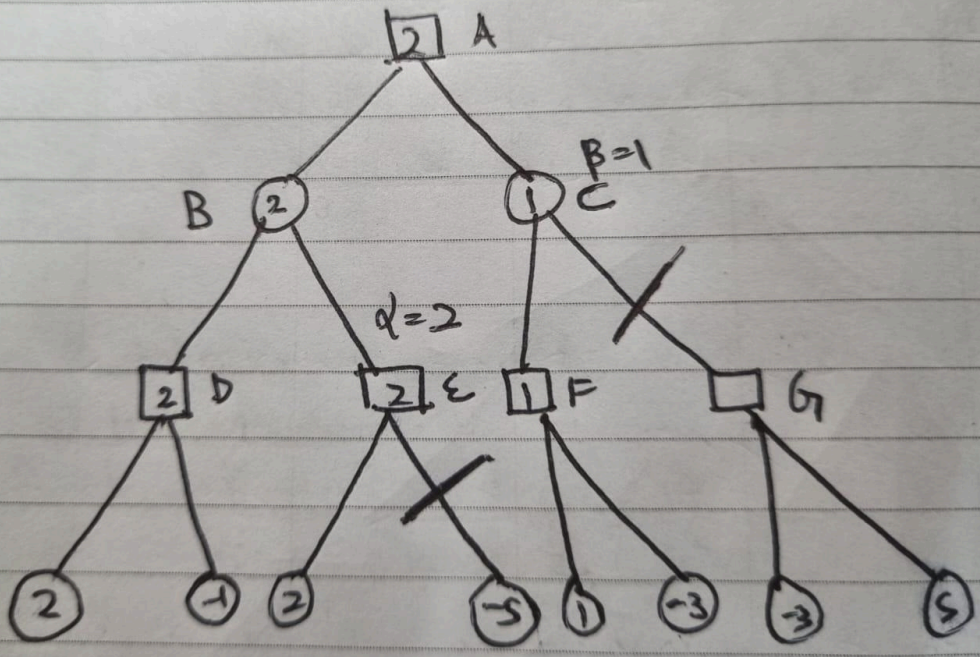
Wk-35, Day 245 - Left 121

Labor Day (US)

08

09

- **Alpha - Beta Pruning**



10

11

12

01

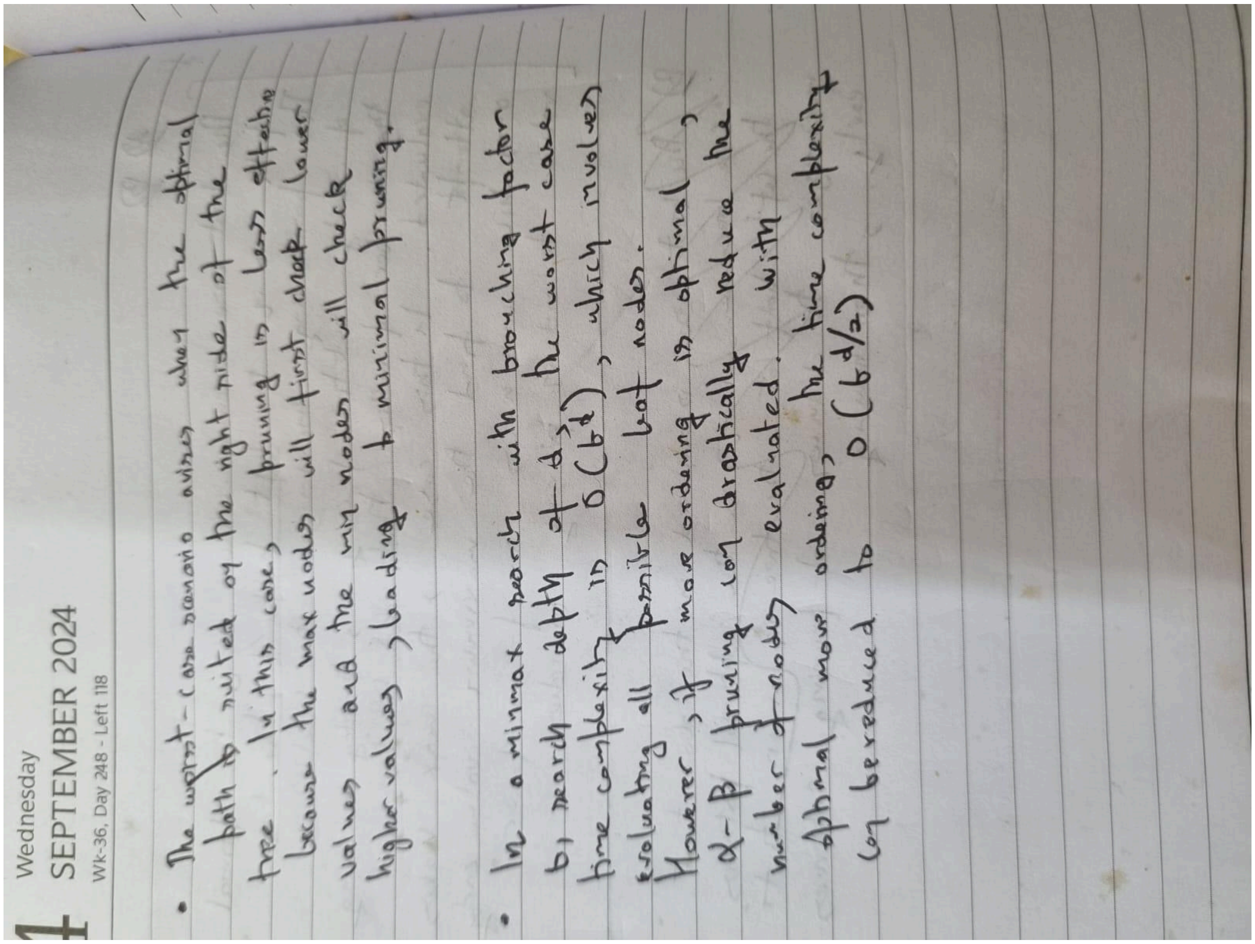
02

Tuesday
SEPTEMBER 2024 3
Wk-36, Day 247 - Left 119

12 ~~A~~ B)

The best case scenario occurs when the optimal path is in the <u>left half of the tree</u>. This positioning maximizes the likelihood of early subtree pruning, thereby reducing the number of nodes that need to be evaluated. In this case, the max nodes attempt to find the maximum value early on, while min nodes aim for a lower value to aid in pruning branches.

B) ~~The best case~~ is achieved in the above test



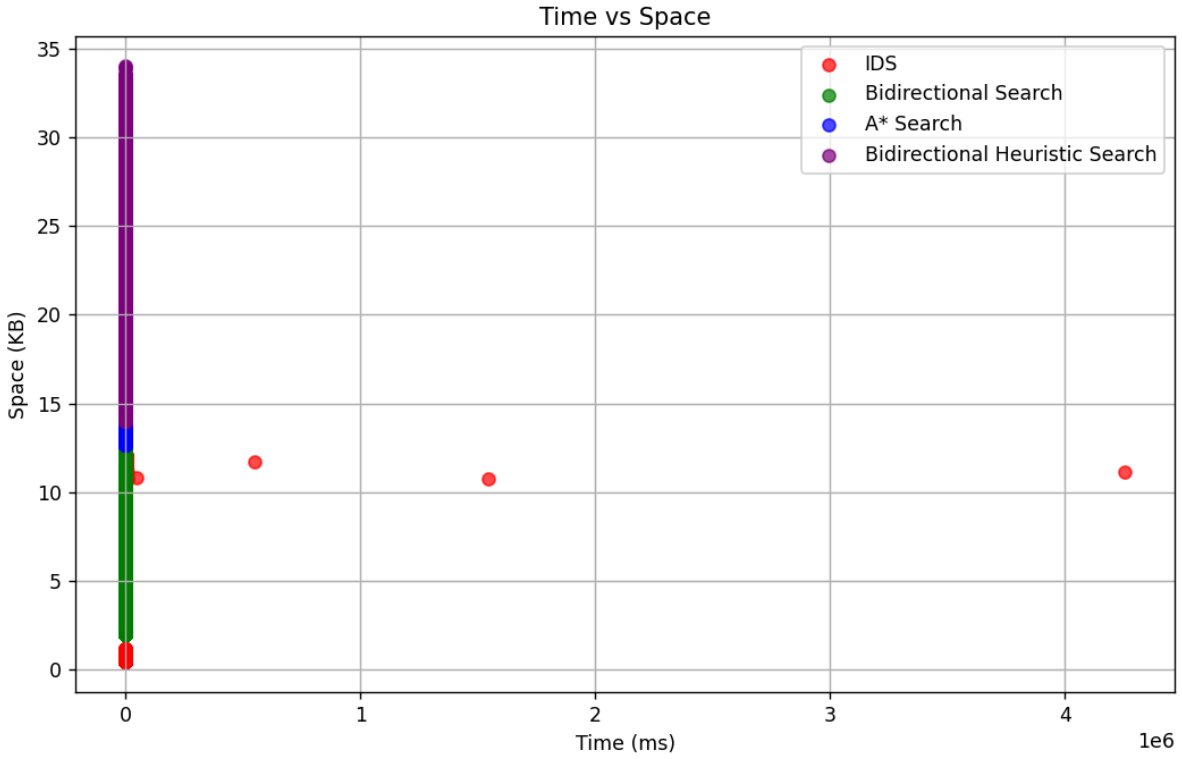
4 Wednesday SEPTEMBER 2024 Wk-36, Day 248 - Left 118

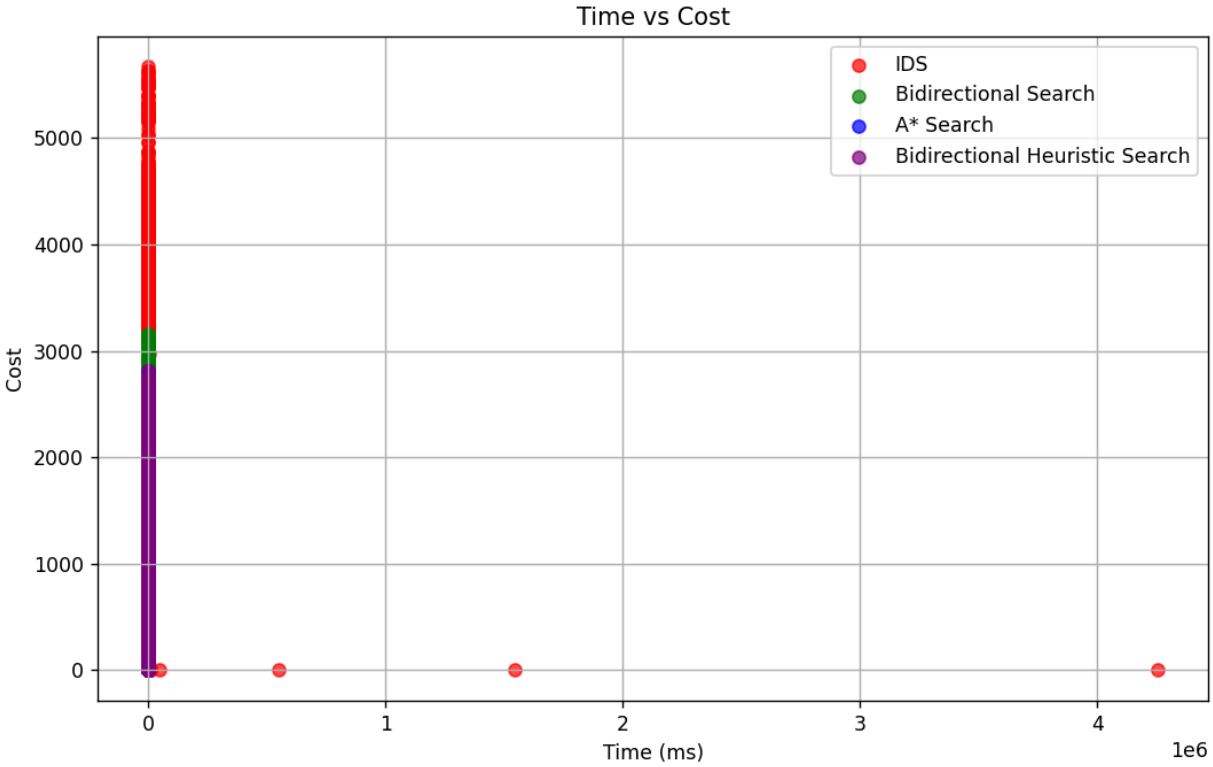
- The worst-case scenario arises when the optimal path is situated on the right side of the tree. In this case, pruning is less effective because the max nodes will first check lower values and the min nodes will check higher values, leading to minimal pruning.
- In a minimax search with branching factor b, search depth of d, the worst case time complexity is $O(b^d)$, which involves evaluating all possible leaf nodes. However, if move ordering is optimal, α-β pruning can drastically reduce the number of nodes evaluated. With optimal move ordering, the time complexity can be reduced to $O(b^{d/2})$

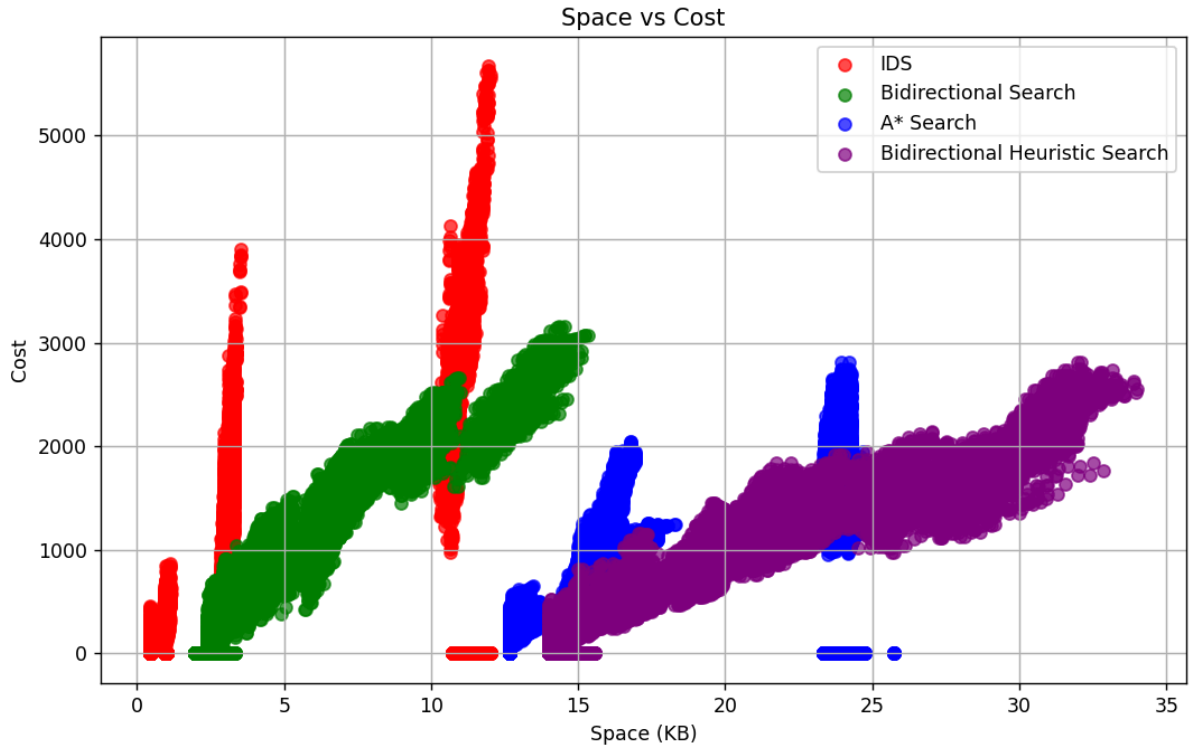
Question 3

b) The paths found by the algorithms can be different for some test cases . The reason is that since IDS implements a depth first search essentially meaning it will explore as far down a branch as possible while in Bidirectional BFS it simultaneously searches from both the start node and the goal node and may return the smallest path in unweighted graph .

c) I plotted the graphs for all the 4 algorithms as following :


Time vs Space
IDS
Bidirectional Search
A* Search
Bidirectional Heuristic Search
Space (KB)
Time (ms)
1e6


Time vs Cost
IDS
Bidirectional Search
A* Search
Bidirectional Heuristic Search
Cost
Time (ms)
1e6



As we can see IDS takes more time in general for finding a path but bidirectional bfs uses more space in general .

e) We find that path found by both the algorithms are not the same because bidirectional A* may return a different path if found optimal from the end as well because it simultaneously explores nodes from both ends while A* expands its frontier along a singular direction

Space : Bidirectional Heuristic Search takes more space than A* on average
Cost : On average they estimate the same optimal paths as can be seen in the graph .

Benefits of Informed Search Algorithms(A* , BFS)

1. Informed algorithms find paths with lower costs as can be seen in cost vs space(3rd) graph .
2. They have lower run time than IDS .

Drawbacks of Informed Search Algorithms

1) The performance is dependent on the choice of heuristic .
2) Bidirectional Heuristic Search takes much more memory on average than other algorithms .