1) Dijkstra's Algorithm

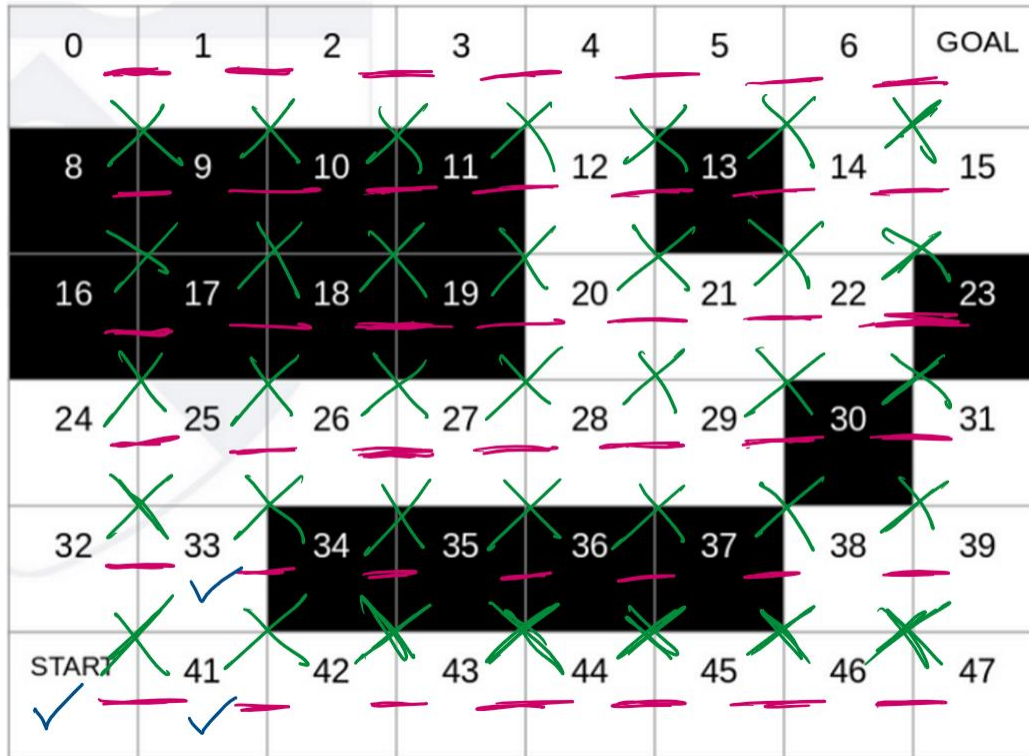| 0	1	2	3	4	5	6	GOAL
8	9	10	11	12	13	14	15
16	17	18	19	20	21	22	23
24	25	26	27	28	29	30	31
32	33	34	35	36	37	38	39
START	41	42	43	44	45	46	47

X - Diagonal paths

— - - Horizontal path

Iteration: 1

Visited nodes so far are [40]

Values : Costs in open set for iteration 1 are [32 :1 , 33 : 1.5, 41 : 1]

Iteration: 2

Visited nodes so far are [32, 33, 40, 41]

Values : Costs in open set for iteration 2 are [33 : 1.5, 41 : 1, 24 : 2, 25 : 2.5]

Iteration: 3

Visited nodes so far are [24, 25, 32, 33, 40, 41]

Values : Costs in open set for iteration 3 are [41 : 1, 24 : 2, 25 : 2.5, 26 : 3.0, 42 : 3.0]

Iteration: 4

Visited nodes so far are [24, 25, 26, 32, 33, 40, 41, 42]

Values : Costs in open set for iteration 4 are [24 : 2, 25 : 2.5, 26 : 3.0, 42 : 2]

Iteration: 5

Visited nodes so far are [24, 25, 26, 32, 33, 40, 41, 42]

Values : Costs in open set for iteration 5 are [25 : 2.5, 26 : 3.0, 42 : 2]

Iteration: 6

Visited nodes so far are [24, 25, 26, 32, 33, 40, 41, 42]

Values : Costs in open set for iteration 6 are [26: 3.0, 42: 2]

Iteration: 7

Visited nodes so far are [24, 25, 26, 32, 33, 40, 41, 42]

Values : Costs in open set for iteration 7 are [42 : 2, 27 : 4.0]

Iteration: 8

Visited nodes so far are [24, 25, 26, 27, 32, 33, 40, 41, 42]

Values : Costs in open set for iteration 8 are [27 : 4.0, 43 : 3]

Iteration: 9

Visited nodes so far are [24, 25, 26, 27, 32, 33, 40, 41, 42, 43]

Values : Costs in open set for iteration 9 are [43 : 3, 20 : 5.5, 28 : 5.0]

Iteration: 10

Visited nodes so far are [20, 24, 25, 26, 27, 28, 32, 33, 40, 41, 42, 43]

Values : Costs in open set for iteration 10 are [20 : 5.5, 28 : 5.0, 44 : 4]

Iteration: 11

Visited nodes so far are [20, 24, 25, 26, 27, 28, 32, 33, 40, 41, 42, 43, 44]

Values : Costs in open set for iteration 11 are [28 : 5.0, 44 : 4, 12 : 6.5, 21 : 6.5, 29 : 7.0]

Iteration: 12

Visited nodes so far are [12, 20, 21, 24, 25, 26, 27, 28, 29, 32, 33, 40, 41, 42, 43, 44]

Values : Costs in open set for iteration 12 are [44 : 4, 12 : 6.5, 21 : 6.5, 29 : 6.0]

Iteration: 13

Visited nodes so far are [12, 20, 21, 24, 25, 26, 27, 28, 29, 32, 33, 40, 41, 42, 43, 44]

Values : Costs in open set for iteration 13 are [12 : 6.5, 21 : 6.5, 29 : 6.0, 45 : 5]

Iteration: 14

Visited nodes so far are [12, 20, 21, 24, 25, 26, 27, 28, 29, 32, 33, 40, 41, 42, 43, 44, 45]

Values : Costs in open set for iteration 14 are [21 : 6.5, 29 : 6.0, 45 : 5, 3 : 8.0, 4 : 7.5, 5 : 8.0]

Iteration: 15

Visited nodes so far are [3, 4, 5, 12, 20, 21, 24, 25, 26, 27, 28, 29, 32, 33, 40, 41, 42, 43, 44, 45]

Values : Costs in open set for iteration 15 are [29 : 6.0, 45 : 5, 3 : 8, 4 : 7.5, 5 : 8.0, 14 : 8.0, 22 : 7.5]

Iteration: 16

Visited nodes so far are [3, 4, 5, 12, 14, 20, 21, 22, 24, 25, 26, 27, 28, 29, 32, 33, 40, 41, 42, 43, 44, 45]

Values : Costs in open set for iteration 16 are [45 : 5, 3 : 8, 4 : 7.5, 5 : 8, 14 : 8, 22 : 7.5, 38 : 7.5]

Visited nodes so far are [3, 4, 5, 12, 14, 20, 21, 22, 24, 25, 26, 27, 28, 29, 32, 33, 38, 40, 41, 42, 43, 44, 45]

Values : Costs in open set for iteration 17 are [3 : 8, 4 : 7.5, 5 : 8, 14 : 8, 22 : 7.5, 38 : 6.5, 46 : 6]

Visited nodes so far are [3, 4, 5, 12, 14, 20, 21, 22, 24, 25, 26, 27, 28, 29, 32, 33, 38, 40, 41, 42, 43, 44, 45, 46]

Values : Costs in open set for iteration 18 are [4 : 7.5, 5 : 8.0, 14 : 8.0, 22 : 7.5, 38 : 6.5, 46 : 6, 2 : 9]

Visited nodes so far are [2, 3, 4, 5, 12, 14, 20, 21, 22, 24, 25, 26, 27, 28, 29, 32, 33, 38, 40, 41, 42, 43, 44, 45, 46]

Values : Costs in open set for iteration 19 are [5: 8, 14 : 8, 22 : 7.5, 38 : 6.5, 46 : 6, 2 : 9]

Visited nodes so far are [2, 3, 4, 5, 12, 14, 20, 21, 22, 24, 25, 26, 27, 28, 29, 32, 33, 38, 40, 41, 42, 43, 44, 45, 46]

Values : Costs in open set for iteration 20 are [14 : 8, 22 : 7.5, 38 : 6.5, 46 : 6, 2 : 9, 6 : 9]

Visited nodes so far are [2, 3, 4, 5, 6, 12, 14, 20, 21, 22, 24, 25, 26, 27, 28, 29, 32, 33, 38, 40, 41, 42, 43, 44, 45, 46]

Values : Costs in open set for iteration 21 are [22 : 7.5, 38 : 6.5, 46 : 6, 2 : 9, 6 : 9, 7 : 9.5, 15 : 9]

Visited nodes so far are [2, 3, 4, 5, 6, 7, 12, 14, 15, 20, 21, 22, 24, 25, 26, 27, 28, 29, 32, 33, 38, 40, 41, 42, 43, 44, 45, 46]

Values : Costs in open set for iteration 22 are [38: 6.5, 46 : 6, 2 : 9, 6 : 9, 7 : 9.5, 15 : 9, 31 : 9]

Visited nodes so far are [2, 3, 4, 5, 6, 7, 12, 14, 15, 20, 21, 22, 24, 25, 26, 27, 28, 29, 31, 32, 33, 38, 40, 41, 42, 43, 44, 45, 46]

Values : Costs in open set for iteration 23 are [46 : 6, 2 : 9, 6 : 9, 7 : 9.5, 15 : 9, 31 : 8, 39 : 7.5, 47 : 8]

Visited nodes so far are [2, 3, 4, 5, 6, 7, 12, 14, 15, 20, 21, 22, 24, 25, 26, 27, 28, 29, 31, 32, 33, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47]

Values : Costs in open set for iteration 24 are [2 : 9, 6 : 9, 7 : 9.5, 15 : 9, 31 : 8, 39 : 7.5, 47 : 7]

Visited nodes so far are [2, 3, 4, 5, 6, 7, 12, 14, 15, 20, 21, 22, 24, 25, 26, 27, 28, 29, 31, 32, 33, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47]

Values : Costs in open set for iteration 25 are [6 : 9, 7 : 9.5, 15 : 9, 31 : 8, 39 : 7.5, 47 : 7, 1 : 10]

Visited nodes so far are [1, 2, 3, 4, 5, 6, 7, 12, 14, 15, 20, 21, 22, 24, 25, 26, 27, 28, 29, 31, 32, 33, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47]

Values : Costs in open set for iteration 26 are [7 : 9.5, 15 : 9, 31 : 8, 39 : 7.5, 47 : 7, 1 : 10]

Visited nodes so far are [1, 2, 3, 4, 5, 6, 7, 12, 14, 15, 20, 21, 22, 24, 25, 26, 27, 28, 29, 31, 32, 33, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47]

Values : Costs in open set for iteration 27 are [15 : 9, 31 : 8, 39 : 7.5, 47 : 7, 1 : 10]

Visited nodes so far are [1, 2, 3, 4, 5, 6, 7, 12, 14, 15, 20, 21, 22, 24, 25, 26, 27, 28, 29, 31, 32, 33, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47]

Values : Costs in open set for iteration 28 are [31 : 8, 39 : 7.5, 47 : 7, 1 : 10]

Parent 31

Visited nodes so far are [1, 2, 3, 4, 5, 6, 7, 12, 14, 15, 20, 21, 22, 24, 25, 26, 27, 28, 29, 31, 32, 33, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47]

Values : Costs in open set for iteration 29 are [39 : 7.5, 47 : 7, 1 :10]

Visited nodes so far are [1, 2, 3, 4, 5, 6, 7, 12, 14, 15, 20, 21, 22, 24, 25, 26, 27, 28, 29, 31, 32, 33, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47]

Values : Costs in open set for iteration 30 are [47 : 7, 1 : 10]

Visited nodes so far are [1, 2, 3, 4, 5, 6, 7, 12, 14, 15, 20, 21, 22, 24, 25, 26, 27, 28, 29, 31, 32, 33, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47]

Values : Costs in open set for iteration 31 are [1 : 10]

Visited nodes so far are [1, 2, 3, 4, 5, 6, 7, 12, 14, 15, 20, 21, 22, 24, 25, 26, 27, 28, 29, 31, 32, 33, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47]

**Values : Costs** in open set for iteration 32 are [0 : 11.0]

Visited nodes so far are [0, 1, 2, 3, 4, 5, 6, 7, 12, 14, 15, 20, 21, 22, 24, 25, 26, 27, 28, 29, 31, 32, 33, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47]

Values : Costs in open set for iteration 33 are []

Final Record: {0: [11.0, 1], 1: [10.0, 2], 2: [9.0, 3], 3: [8.0, 12], 4: [7.5, 12], 5: [8.0, 12], 6: [9.0, 14], 7: [9.5, 14], 12: [6.5, 20], 14: [8.0, 21], 15: [9.0, 22], 20: [5.5, 27], 21: [6.5, 28], 22: [7.5, 29], 24: [2, 32], 25: [2.5, 33], 26: [3.0, 33], 27: [4.0, 26], 28: [5.0, 27], 29: [6.0, 28], 31: [8.0, 38], 32: [1, 40], 33: [1.5, 40], 38: [6.5, 45], 39: [7.5, 46], 40: [0, None], 41: [1, 40], 42: [2, 41], 43: [3, 42], 44: [4, 43], 45: [5, 44], 46: [6, 45], 47: [7, 46]}

Final Path:
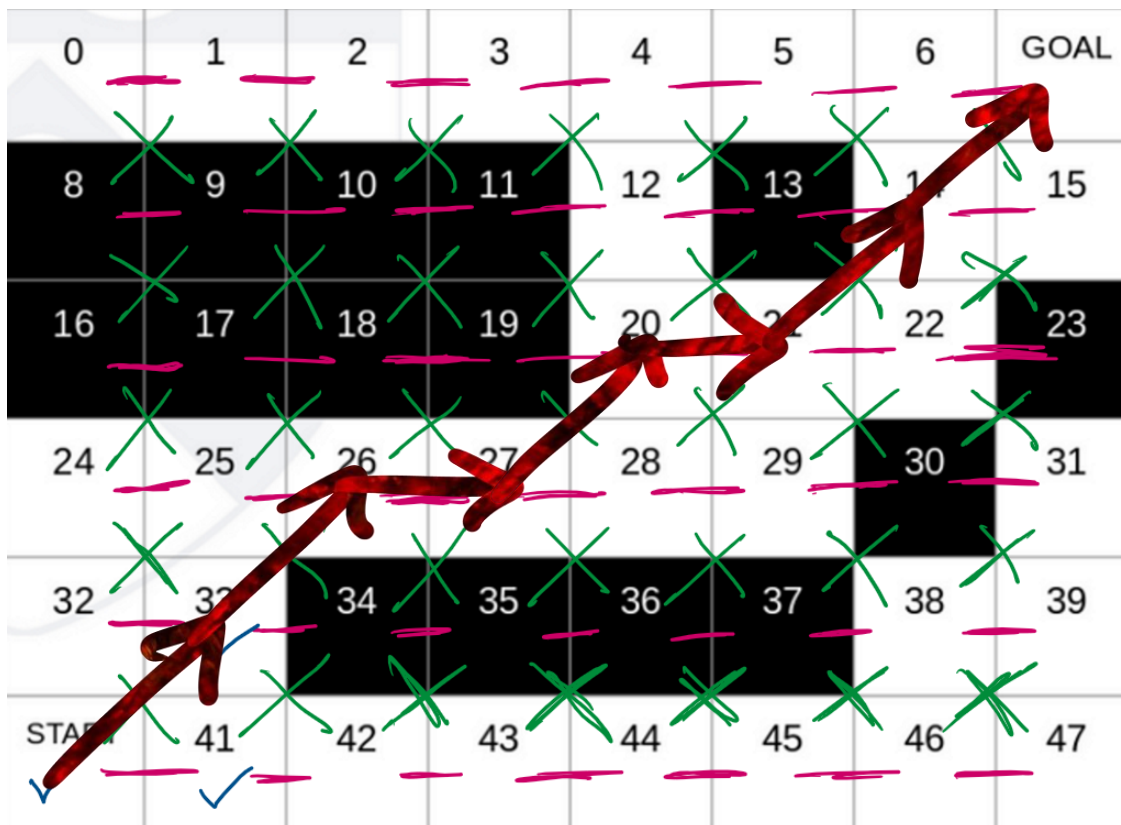
2) A* Algorithm

10 0 7	9 1 6	8 2 5	7 3 4	7 4 3	7 5 2	7 6 1	GOAL
17	15	13	11	10	9	8	
8	9	10	11	6 12 4 / 10	13	6 14 2	7 15 1 / 8
16	17	18	19	5 20 5 / 10	5 21	6 22 3 / 9	23
2 24 10 / 12	2 25 9 / 11	2 26 8 / 10	3 27 7 / 10	4 28 / 10	5 29 5 / 10	30	7 31 3 / 10
1 32 11 / 12	1 33 10 / 1	34	35	36	37	6 38 5 / 11	7 39 4 / 11
START	1 41 11 / 12	2 42 10 / 12	3 43 9 / 12	4 44 8 / 12	5 45 7 / 12	6 46 6 / 12	7 47 5 / 12

- Heuristic distance
- Start distance
- Addition of Both

Red is estimated distance between node and Goal (H)
Green is distance to reach the node from the start (G)
Black is the combined cost for each (F)

$$F = G + H \rightarrow goal - node$$
$$\downarrow$$
$$start - node$$

At every step: the algorithm will assign costs and the optimal path is ↓
From start node, we go towards the node with least cost function.
Until, we reach the goal.

For example

Node 14 : $G = 6$ $H = 2$

$$F = G + H$$
$$= 6 + 2$$
$$= 8$$

∴ From node 21 , we choose to go to node 14 (cost = 8) and not node 22 (cost = 9) or node 20 (cost = 10)

3) Difference between A* and Dijkstra's

- A* uses a heuristic function to calculate the distance between a node and goal.
  We don't need this estimation in Dijkstra's algorithm
- Time complexity of Dijkstra's is more

4) Data Structure for A* and Dijkstra's

- Priority queue : Used to store the open set in both A* and Dijstra's

5) Which algorithm returns an optimal path ?

- **Dijkstra's** is guaranteed to return the optimal path in any graph except for negative weight. But, A*, because it uses an heuristic function, it might not return the optimal path always. It depends on heuristic function correctness.