9414 Asst2

z5102511 - Hao Fu

Question 1 - Search Algorithms for the 15-Puzzle

(a)

	Start10	Start20	Start27	Start35	Start43
UCS	N=2565	Mem	Mem	Mem	Mem
IDS	G=10, N=2407	G=20, N=5297410	Time	Time	Time
A*	G=10, N=33	N=915	G=27,N=1873	Mem	Mem
IDA*	G=10. N=29	G=20, N=952	G=27,N=2237	G=35, N=215612	G=43, N=2884650

(b)

- UCS: It is the most naive algorithm among four algorithm, it had most time and memory consumption.
- IDS: It require less memory than UCS, but it still pretty slow. Because the number of node is pretty large.
- A*: It shows more efficient than UCS and IDS. The number of node decrease dramatically. But it still need a large memory to run.
- IDA*: The most memory and time efficient among four algorithm. It require smallest memory, but it used less time.

Question 2 - Deceptive Starting States

(a)

Result	Start49	Start51
State	MBDC LAKH JFEI ONG	GKJI MNC EOHA FBLD
heuristic	25	43

(b)

	Start51
Number of Node	551168

(c)

Beacause IDA* is a memory efficient algorithm, it will not store the middle status. If the path is longer than our goal, then it will start from root node. Thus, it will access some node a lot of times. G=49 is close to G=51. But it need many try to reach our goal which is path length less or equal to 49.

Question 2 - Heuristic Path Search

(a)

		Start49		**Start60**		**Start64**
IDA*	49	178880187	60	321252368	64	1209086782
1.2						
1.4						
Greedy	133	5237	166	1617	184	2174

(b)

		Start49		**Start60**		**Start64**
IDA*	49	178880187	60	321252368	64	1209086782
1.2	51	988332	62	230861	66	431033
1.4						
Greedy	133	5237	166	1617	184	2174

(c)

		Start49		Start60		Start64
IDA*	49	178880187	60	321252368	64	1209086782
1.2	51	988332	62	230861	66	431033
1.4	57	311704	82	4432	94	190278
Greedy	133	5237	166	1617	184	2174

(d)

As we can find in the table, Greedy algorithm had the faster speed but the longest path. Meanwhile. the IDA* can find the shortest path but it need a lot of time to execute. When we set the $w = 1.2$ the number of node drop dramatically and the result is pretty close to our best result. But if we set $w = 1.4$, the speed is faster but the quality of the result is became worst. In conclusion, $w = 1.2$ is the best tradeoff among four algorithm.

Qestion 3 - Maze Search Heuristics

(a)

Using **Manhattan Distance heuristic**

$$h(x, y, x_G, y_G) = |x - x_G| + |y - y_G|$$

(b)

(i) No, Because it can not estimate the minimum cost.

(ii) No, It cant handle diagonally step, it won't be the minimum cost.

(iii) Chebyshev Distance Heuristic:

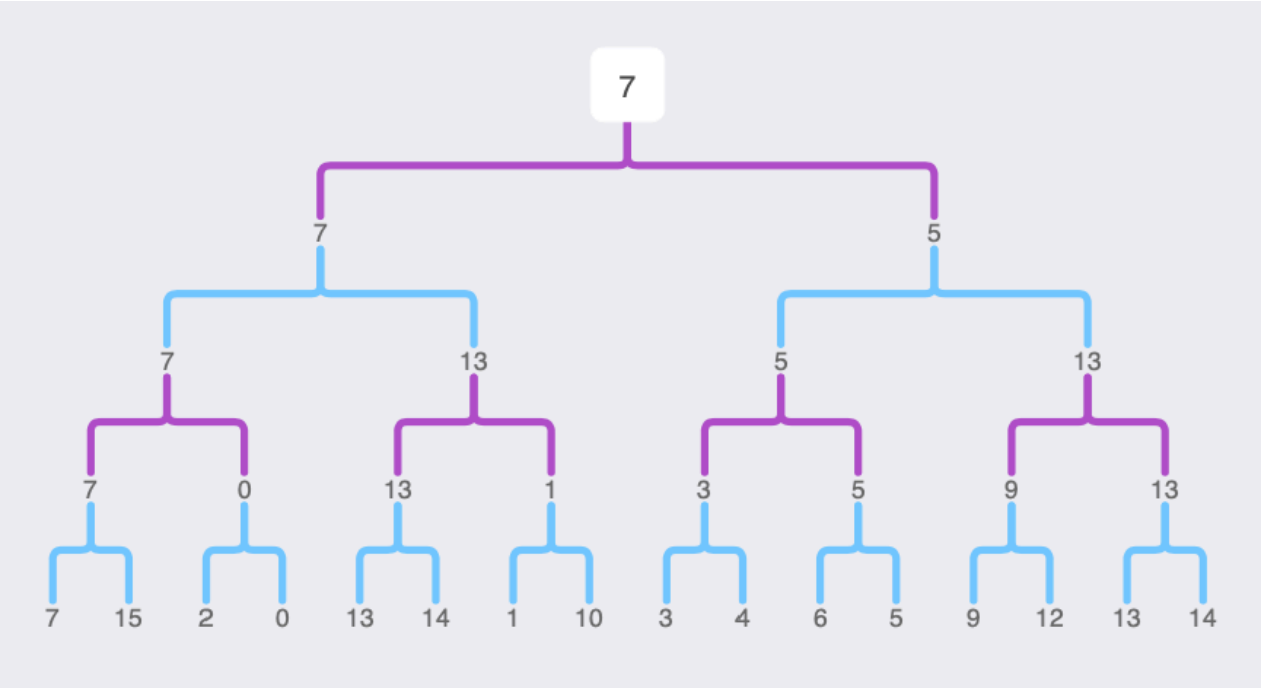
$$h(x, y, x_G, y_G) = max(|x - x_G|, |y - y_G|)$$

Question 4 - Game Trees and Pruning

(a)

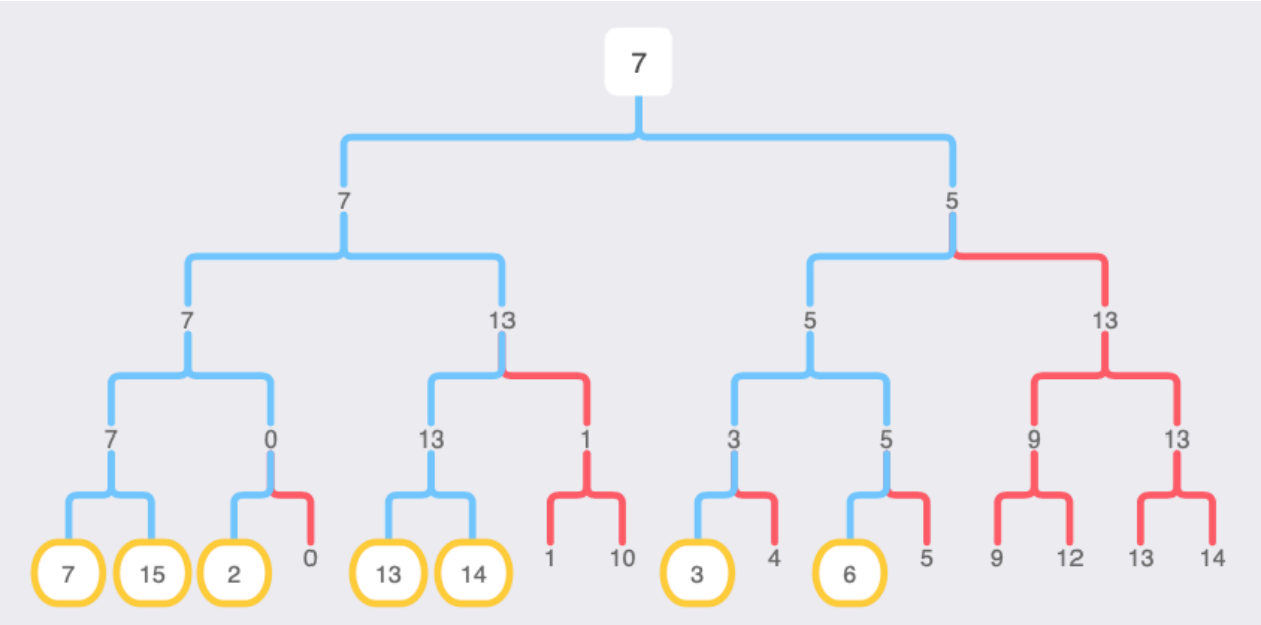
The following tree is one of the example that alpha-beta can prune the most nodes which is **9** nodes.

The blue line represent MIN select medthod, and the purple line represent MAX select medthod.



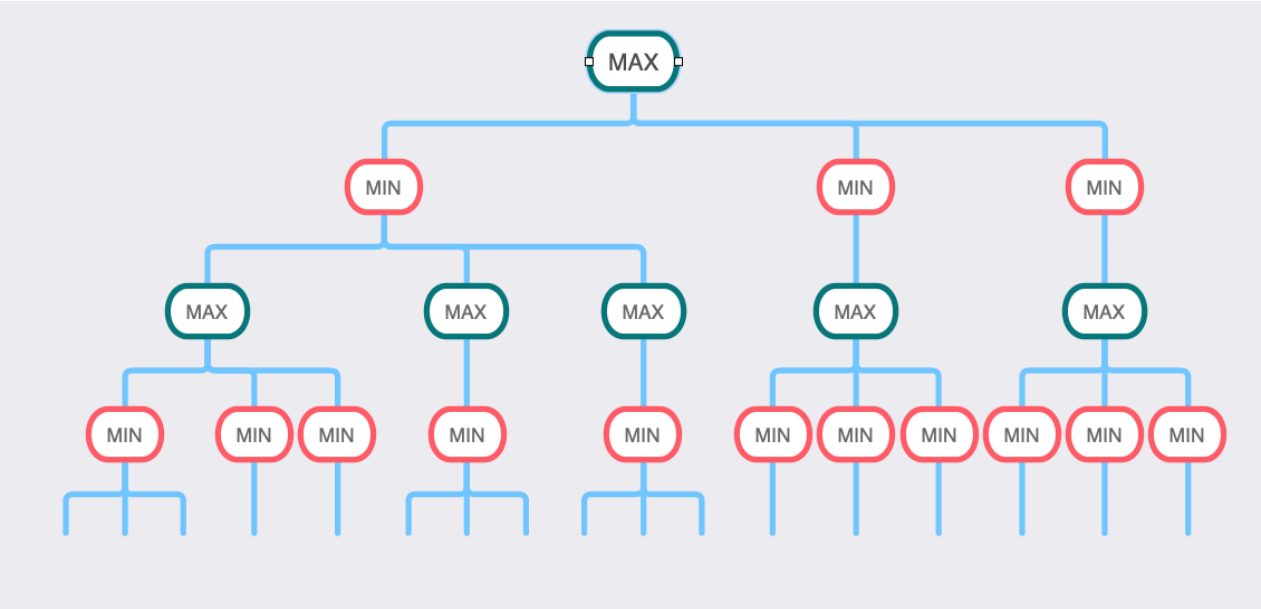
(b)

The node with yellow border is the node accessed. The pruning node and line is shown as **red** line.



(c)

There will be 17 node evaluated and the other 61 node will be pruned.



(d)

The time complexity of alpha-beta alogrithm should be

$$O(b^{\frac{d}{2}})$$

where b is branching factor and d is the depth of the tree.

All the situation can be separated as 2: tree depth is odd and even.

- Odd: The best move is always search first.
- The number of node are evaluated: $O(b * 1 * b * 1 * \ldots * b) = O(b^{\frac{d}{2}})$

- Even: The wrost move is always search first.
- The number of node are evaluated: $O(b * 1 * b * 1 * \ldots * 1) = O(b^{\frac{d}{2}})$

Thus the result is $O(b^{\frac{d}{2}})$.