

Sam Hopkins
CS470
hw 8

1. a). i.

15	1	14	2	3	4	5	6	7	8	9	10	11	12	13
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O

ii. 14

iii. Go down left path towards 14

b) i.

8	7	5	6	10	9	3	4	1	2	15	14	13	12	11
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O

ii. 6

iii. To follow left most path towards ~~14~~ 6

c) i. ~~| | | | | | | | | | | | | | | |
|----|---|----|----|----|----|----|---|---|---|---|---|---|---|---|
| 14 | 1 | 15 | 13 | 12 | 11 | 10 | 1 | 8 | 7 | 6 | 5 | 4 | 3 | 2 |
| A | B | C | D | E | F | G | H | I | J | K | L | M | N | O |~~

ii. ~~14~~ 14

iii. ~~14~~ 14 ^{parent} ^{their parent} ^{parent} ^{their parent} ^{parent}
~~14~~ 14 ^{D, G, H, K, L, M, N, O}

iv. Follow towards ~~14~~ ¹⁴ leftmost path

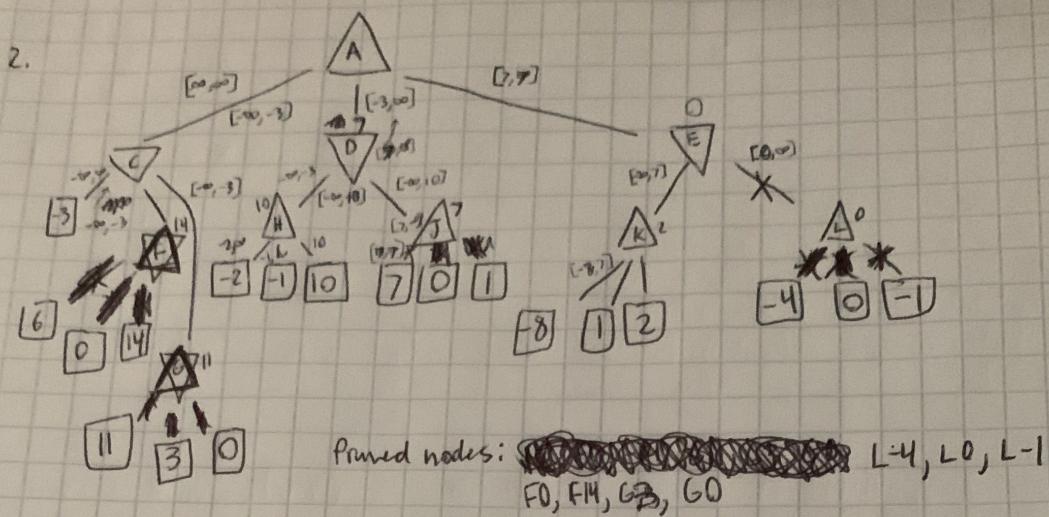
d) i.

6	5	15	14	9	8	7	10	1	2	12	13	4	3	11
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O

ii. 8

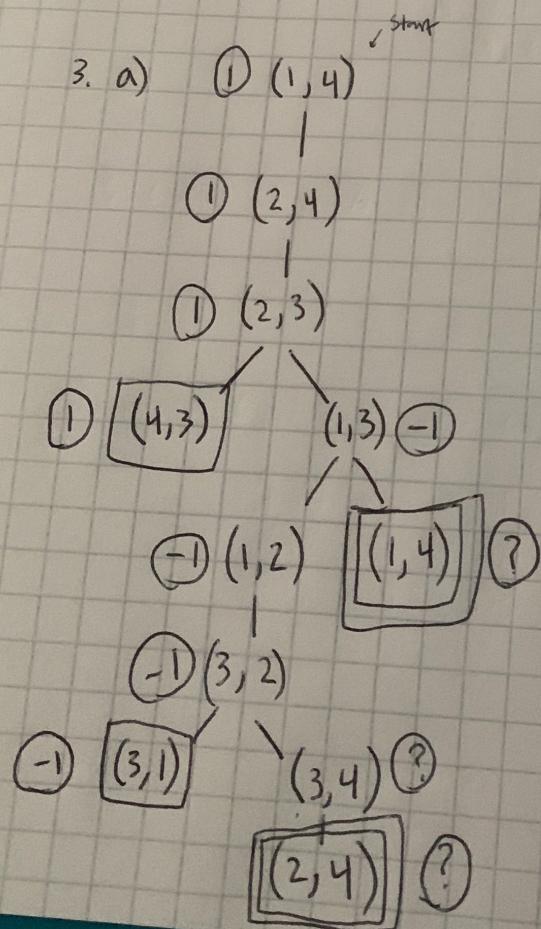
iii. D, K, L, M, N, O

iv. Towards the 8 nodes the middle path



b) 7

c) They should go towards D



b) ? values are given and handled in a way that any node would choose a definite path other than uncertainty, such that any win is better than a loop. If all successors are ?, then that will have to be ?

c) Because min-max is depth first search, it will end up in an infinite loop. We could treat ? as a draw state, or make sure we haven't already visited that state in the same iteration (keep track of the path and check all visited states). It won't work perfectly, and some games will have trees that still won't work with this version of the algorithm.