CS303A Homework 1

Jiang Yuchen 11812419

Q1.

a.

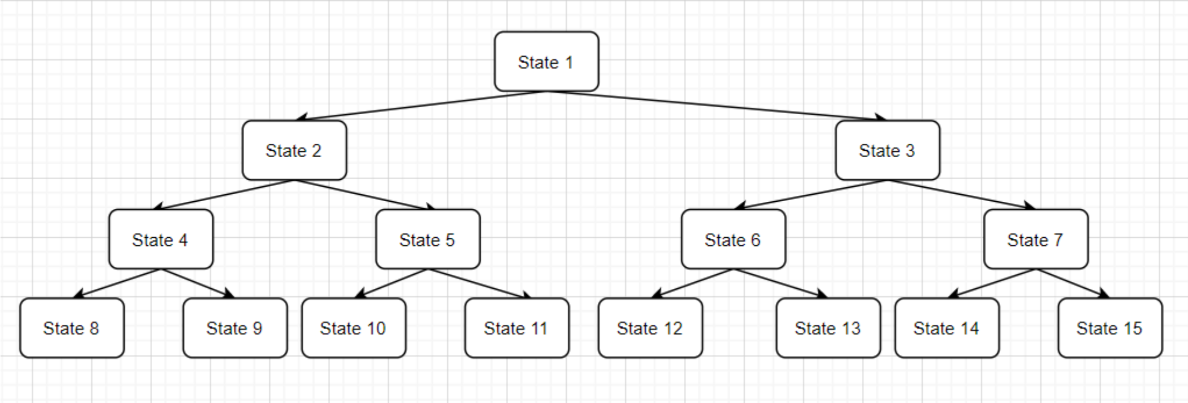


Figure.1

State space shows above for states 1 to 15.

b.

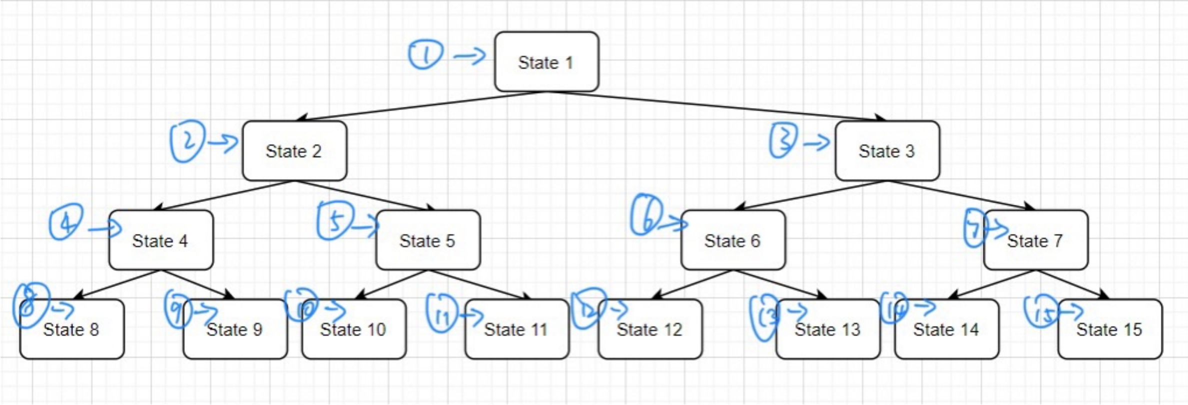


Figure.2

According to Figure.2,we apply BFS to find goal state 11, and the order is:

state 1,state 2,state 3,state 4,state 5,state 6,state 7,state 8,state 9,state 10,state 11.

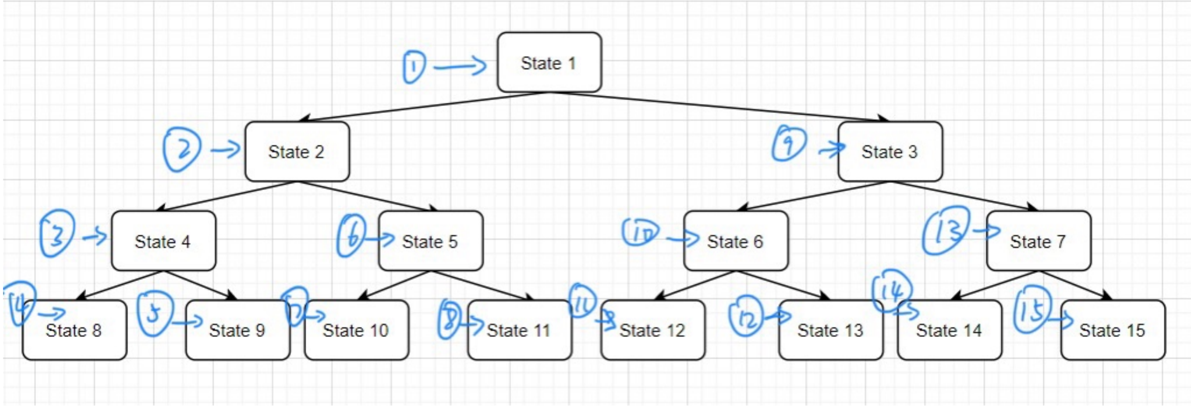


Figure.3

For DFS with depth limit 3(Figure.3), the order is:

state 1,state 2,state 4,state 8,state 9,state 5,state 10, state 11.

For IDS,it will search for four turns. The order is :

Turn 1: state 1.

Turn 2: state 1,state 2,state 3.

Turn 3: state 1,state 2,state 4,state 5,state 3,state 6,state 7.

Turn 4: state 1,state 2,state 4,state 8,state 9,state 5,state 10,state 11.

c.

Bidirectional search works very well on this problem since it only need two turns to finish the task.

Forward direction: 1→[2,3], 2→[4,5]

Backward direction: 11→ [5], 5→[2].

Two graph intersect in two turns. And the nodes used in forward direction is 5,depth is 2. According to the definition of branching factor,b* is 2 for forward direction.

Similarly, nodes is 4 and depth is 2 for backward direction,so b* is 1 for backward direction.

So,branching factor is 2 for forward direction, and 1 for backward direction.

d.

Yes. According to c.,we can find that in backward direction,the next state for state x can only be $\lfloor x/2 \rfloor$,which makes it branching factor 1. Thus,the problem can be considered as traveling from goal state to initial state 1.

e.

According to reformulation in e.,the algorithm is

```
x is initialed as goal state
While x is not 1
Travel from state x to $\lfloor x/2 \rfloor$ and judge whether is Reverse-Left or Reverse-Right and record it
Output order according to Left/Right record
```

Q2.

a.

We can suppose that each city can be visited for more than once and also there is no more cost on one edge when it is visited more than once. Thus we can apply MST algorithm on TSP and get our result.

b.

Since the shortest distance between two cities is straight-line distance ,so if there is an straight line edge between current city and start city,MST heuristic will dominates straight-line distance from the current city back to the start city.

c.

```
Generate an initial route r.
Calculate the distance d for route r.
Do while (True)
   Generate neighbor routes r's based on r
   Calculate the each distance d' for route r'
   If there exist d'<d ,then route r = r',d = d'.
   Otherwise route r is the best answer,break.
```

d.

```
Initial population as a random route r,generation for 100,possibility of mutation is 0.1.
Initial fitness threshold is distance of route r.
Then generate new population based on route r:
   For each child, generate random number to test whether there could be mutation.
   Append child to new population,and take it as initial population.
Return the route with minimum distance in last population.
```

Q3.

a.

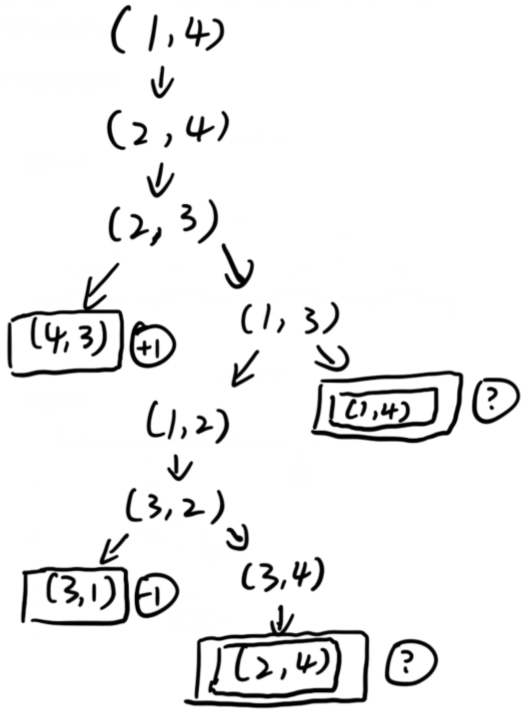


Figure.4

According to Figure.4,the complete game tree shows above.

b.

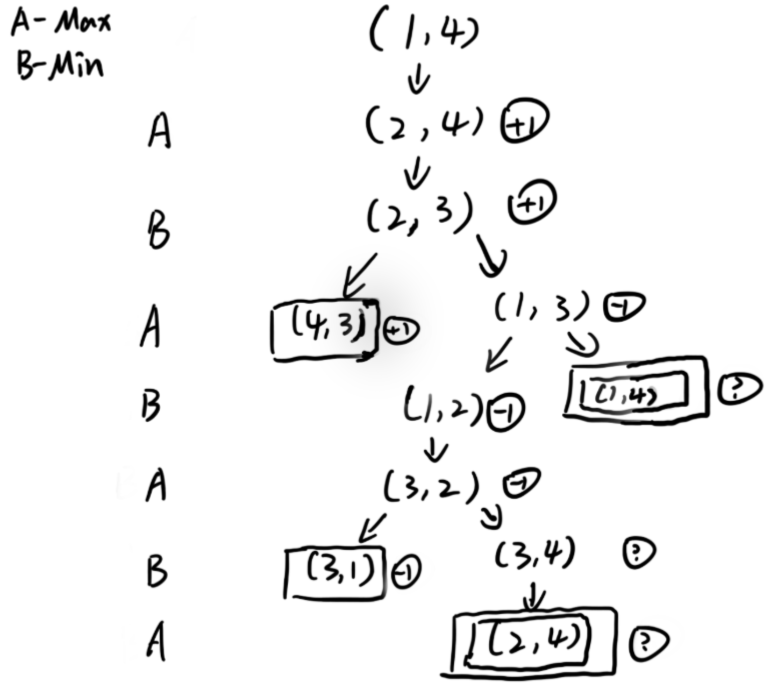
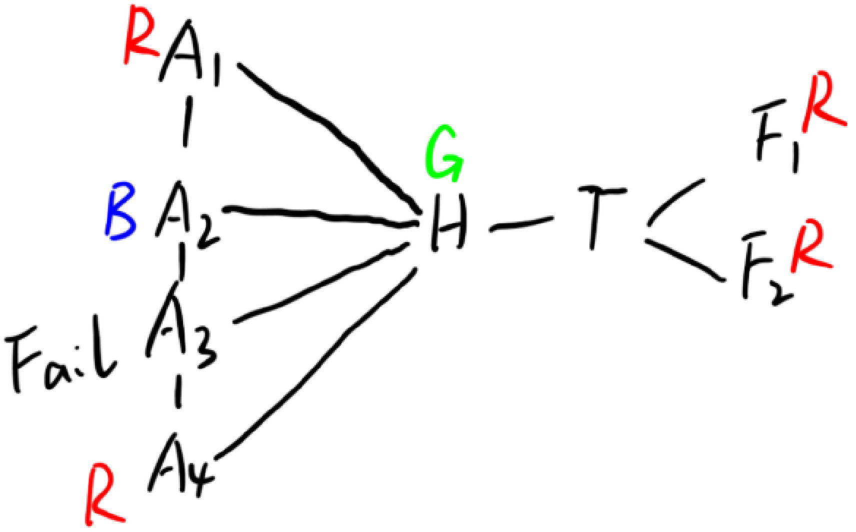


Figure.5

According to Figure.5,minimax values are marked as above. For those nodes which have successor "?",we apply min(-1,?) = -1 and mark their value.

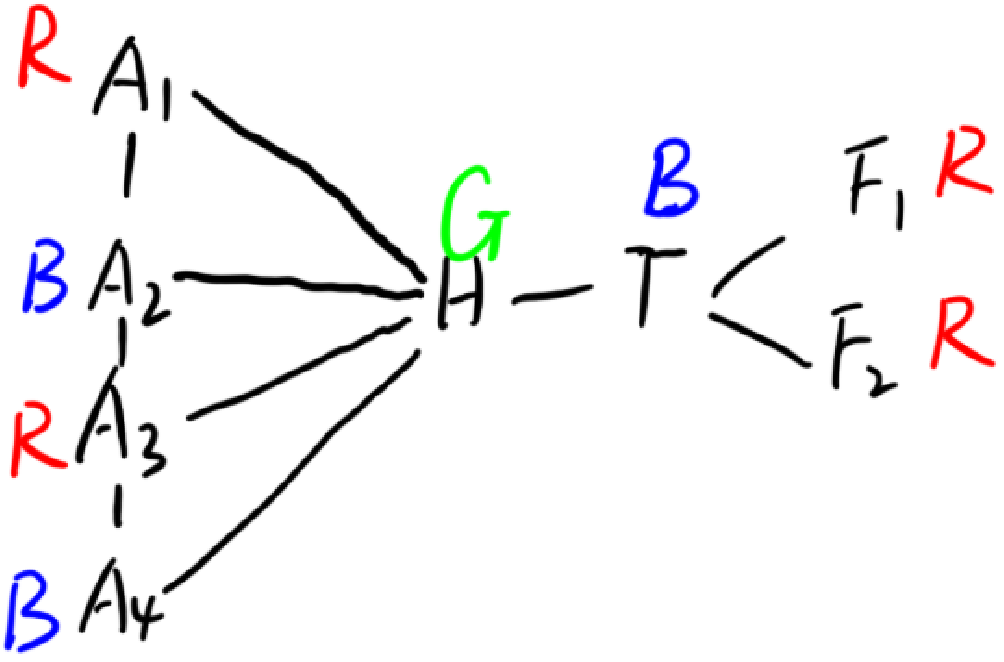
Q4.

A1-R, H-G, A4-R, F1-R, A2-B, F2-R, A3-FAIL



Traceback to A4-R,change it to A4-B

A1-R, H-G,A4-B, F1-R, A2-B, F2-R, A3-R, T-B. Got it!


R A1
B A2
R A3
B A4
G
H
B
T
F1 R
F2 R