Assignment 1 – Search as Puzzle Solving (or Problem Solving)

Puzzles

1. **Cannibals and Missionaries**. There are *n* missionaries and *n* cannibals on the right bank of a river. All of them want to get to the left bank of the river. There is a row boat that can carry no more than *c* number of passengers at a time. However, if the cannibals outnumber the missionaries on the boat or on either river bank, the missionaries will be killed. How can all of them get across the river safely? What is the solution with the fewest crossings?

The code should be able to solve for:

a) n=3 and c=2

b) n=4 and c=3

c) n=5 and c=3

Note: there is no solution for n=4 and c=2

Source: Gardner, M. The Last Recreations – Hydras, Eggs and Other Mathematical Mystifications. Copernicus, Springer-Verlag. 1997.

1. Formal problem representation

M= missionaries、C=cannibals

State representation：[M,M,M,C,C,C],where the missionaries and cannibals are represented by column positions.

Start state: [1,1,1,1,1,1]

Goal state:[0,0,0,0,0,0]

State space: [0,0,0,0,0,0]

Operators: moves= [[0, 1], [0, 2], [0, 3], [0, 4], [0, 5], [1, 2], [1, 3], [1, 4], [1, 5], [2, 3], [2, 4], [2, 5], [3, 4], [3, 5], [4, 5]]。

It means that two people will be sent from the right at a time, depending on the location.

For example, move[1,2] means that people at position 1 and position 2 will be carried over, and state[1,1,1,1,1,1] will become [1,0,0,1,1,1] after move[1,2].

2. Explain the heuristics you have chosen and why it works. Explain also how it is related to the Search strategy.

Heuristic：How many people need to be moved from the current state to the goal state

Cost= How many people need to be moved from the current state to the goal state

e.g. current state [1,1,1,1,1,0] , goal state=[0,0,0,0,0,0],and cost =5

e.g. current state [1,1,1,0,1,0] , goal state=[0,0,0,0,0,0],and cost =4

Using greedy Best-First Search：Expanding the current state and compare the cost of each current state, then choose the smallest cost state to expand. Repeat this process until it finds an answer

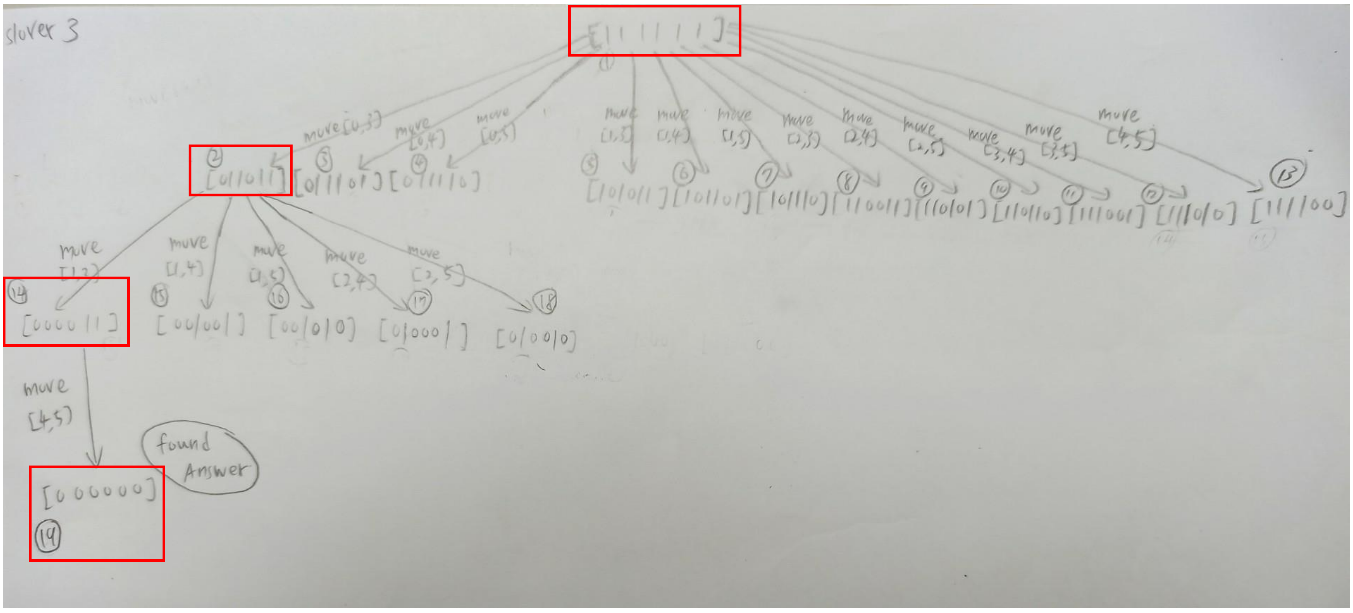
3. Any observations and /or discussion, e.g. how the choice of representation will affect the solving efficiency, etc. Explain the state representation that you have chosen, especially why it is a good minimal representation.

Such a state means that it is easier to perform heuristics, and the cost is easier to calculate

4. Solution that is obtained as a printout from the code. This should be paired with the search tree diagram

```
move  [] ->  [1 1 1 1 1 1]

moves= [[0, 3], [0, 4], [0, 5], [1, 3], [1, 4], [1, 5], [2, 3], [2, 4], [2, 5], [3, 4], [3, 5], [4, 5]]
trying ... [0, 3]
move  [0, 3] ->  [0 1 1 0 1 1]

trying ... [0, 4]
move  [0, 4] ->  [0 1 1 1 0 1]

trying ... [0, 5]
move  [0, 5] ->  [0 1 1 1 1 0]

trying ... [1, 3]
move  [1, 3] ->  [1 0 1 0 1 1]

trying ... [1, 4]
move  [1, 4] ->  [1 0 1 1 0 1]

trying ... [1, 5]
move  [1, 5] ->  [1 0 1 1 1 0]

trying ... [2, 3]
move  [2, 3] ->  [1 1 0 0 1 1]

trying ... [2, 4]
move  [2, 4] ->  [1 1 0 1 0 1]

trying ... [2, 5]
move  [2, 5] ->  [1 1 0 1 1 0]

trying ... [3, 4]
move  [3, 4] ->  [1 1 1 0 0 1]

trying ... [3, 5]
move  [3, 5] ->  [1 1 1 0 1 0]

trying ... [4, 5]
move  [4, 5] ->  [1 1 1 1 0 0]

move  [[0, 3], [0, 4], [0, 5], [1, 3], [1, 4], [1, 5], [2, 3], [2, 4], [2, 5], [3, 4], [3, 5], [4, 5]] ->  [0 1 1 0 1 1]

moves= [[1, 2], [1, 4], [1, 5], [2, 4], [2, 5]]
trying ... [1, 2]
move  [1, 2] ->  [0 0 0 0 1 1]

trying ... [1, 4]
move  [1, 4] ->  [0 0 1 0 0 1]

trying ... [1, 5]
move  [1, 5] ->  [0 0 1 0 1 0]

trying ... [2, 4]
move  [2, 4] ->  [0 1 0 0 0 1]

trying ... [2, 5]
move  [2, 5] ->  [0 1 0 0 1 0]

move  [[1, 2], [1, 4], [1, 5], [2, 4], [2, 5]] ->  [0 0 0 0 1 1]

moves= [[4, 5]]
trying ... [4, 5]
move  [4, 5] ->  [0 0 0 0 0 0]

Found answer
```

5. A drawing of the of the search process as a search tree using the solution given by the printout



2. Three men robbed a gentleman of a vase, containing 24 ounces of balsam. Whilst running away they met a glass seller, of whom they purchased three vessels. On reaching a place of safety they wished to divide the booty, but found that their vessels could hold 5, 11, and 13 ounces respectively. How could they divide the balsam into equal portions?

Source: 1.W. W. Rouse Ball and H. S. M. Coxeter, Mathematical Recreations and Essays, Dover, 1987

1. Formal problem representation

State representation：[A,B,C,D]

A is the number of ounces of balsam in the 24 ounces vase

B is the number of ounces of balsam in the 13 ounces vessels

C is the number of ounces of balsam in the 11 ounces vessels

D is the number of ounces of balsam in the 5 ounces vessels

Start state: [24,0,0,0]

Goal state: [8,8,8,0]

Operators: moves：

('A', 'B') ('A', 'C') ('A', 'D')

('B', 'A') ('B', 'C') ('B', 'D')

('C', 'A') ('C', 'B') ('C', 'D')

('D', 'A') ('D', 'B') ('D', 'C')

e.g. ('A','D') means pour from A to D. A must fill D or fill it until A is empty.

2. Explain the heuristics you have chosen and why it works. Explain also how it is related to the Search strategy.

Heuristics=Calculate how many differences between the capacity of each container in the current state and the capacity of each container in the target state.

Cost= how many differences between the capacity of each container in the current state and the capacity of each container in the target state.

e.g. current state=[0,13,11,0],goal state=[8,8,8,0], and cost=3

e.g. current state = [16,0,8,0] , goal state=[8,8,8,0] , and cost=2

Using greedy Best-First Search ： Expanding the current state and compare the cost value of each current state, then choose the smallest cost state to expand. Repeat this process until it finds an answer.

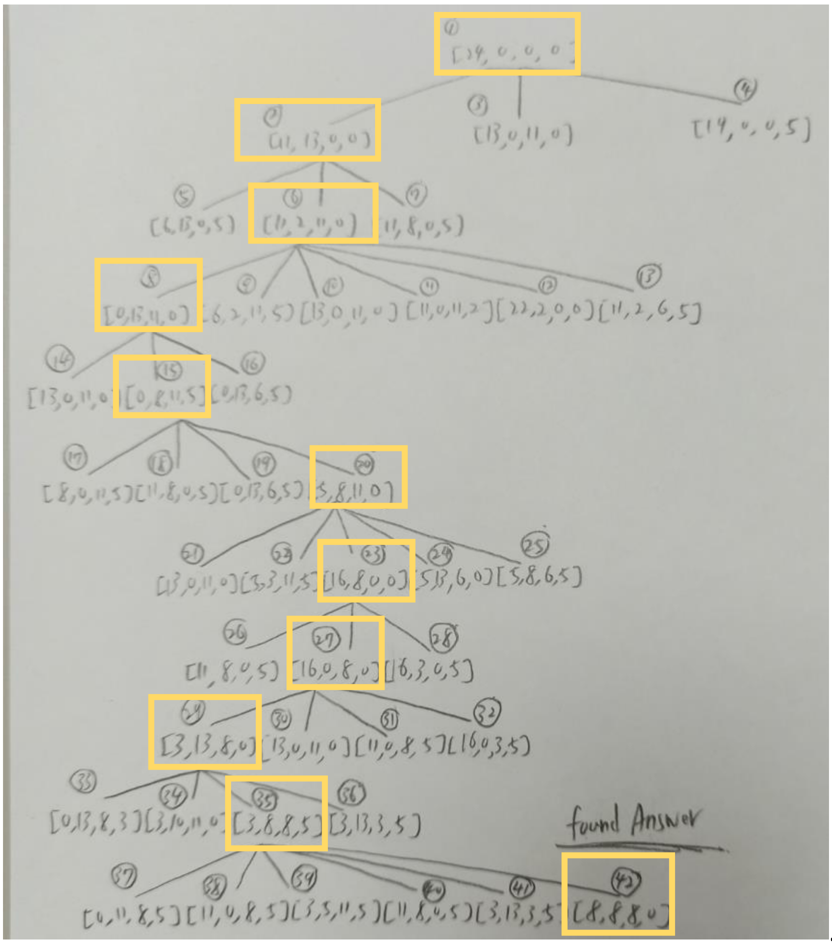
3. Any observations and /or discussion, e.g. how the choice of representation will affect the solving efficiency, etc. Explain the state representation that you have chosen, especially why it is a good minimal representation.

Such a state means that it is easier to perform heuristics, and the cost is easier to calculate

4. Solution that is obtained as a printout from the code. This should be paired with the search tree diagram

```
getMoves for ..  [24, 0, 0, 0]
--->moved  [11, 13, 0, 0]
--->moved  [13, 0, 11, 0]
--->moved  [19, 0, 0, 5]
getMoves for ..  [11, 13, 0, 0]
--->moved  [6, 13, 0, 5]
--->moved  [11, 2, 11, 0]
--->moved  [11, 8, 0, 5]
getMoves for ..  [11, 2, 11, 0]
--->moved  [0, 13, 11, 0]
--->moved  [6, 2, 11, 5]
--->moved  [13, 0, 11, 0]
--->moved  [11, 0, 11, 2]
--->moved  [22, 2, 0, 0]
--->moved  [11, 2, 6, 5]
getMoves for ..  [0, 13, 11, 0]
--->moved  [13, 0, 11, 0]
--->moved  [0, 8, 11, 5]
--->moved  [0, 13, 6, 5]
getMoves for ..  [0, 8, 11, 5]
--->moved  [8, 0, 11, 5]
--->moved  [11, 8, 0, 5]
--->moved  [0, 13, 6, 5]
--->moved  [5, 8, 11, 0]
getMoves for ..  [5, 8, 11, 0]
--->moved  [13, 0, 11, 0]
--->moved  [5, 3, 11, 5]
--->moved  [16, 8, 0, 0]
--->moved  [5, 13, 6, 0]
--->moved  [5, 8, 6, 5]
getMoves for ..  [16, 8, 0, 0]
--->moved  [11, 8, 0, 5]
--->moved  [16, 0, 8, 0]
--->moved  [16, 3, 0, 5]
getMoves for ..  [16, 0, 8, 0]
--->moved  [3, 13, 8, 0]
--->moved  [13, 0, 11, 0]
--->moved  [11, 0, 8, 5]
--->moved  [16, 0, 3, 5]
getMoves for ..  [3, 13, 8, 0]
--->moved  [0, 13, 8, 3]
--->moved  [3, 10, 11, 0]
--->moved  [3, 8, 8, 5]
--->moved  [3, 13, 3, 5]
getMoves for ..  [3, 8, 8, 5]
--->moved  [0, 11, 8, 5]
--->moved  [11, 0, 8, 5]
--->moved  [3, 5, 11, 5]
--->moved  [11, 8, 0, 5]
--->moved  [3, 13, 3, 5]
--->moved  [8, 8, 8, 0]
Found answer
```

5.A drawing of the of the search process as a search tree using the solution given by the printout



.