CSC 404 - Activity/Project 1 - Name:

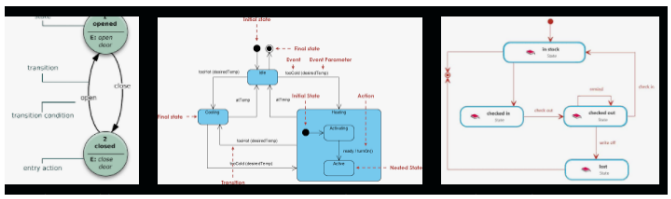
Problem 1. On one bank of a river are **five** missionaries and **five** cannibals. There is one boat available that can hold up to **three** people and that they would like to use to cross the river. If the cannibals ever outnumber the missionaries on either of the river's banks, the missionaries will get eaten. How can the boat be used to safely carry all the missionaries and cannibals across the river? (Note - we also cannot have 2 cannibals and 1 missionary in a boat together.)

a. Solve the riddle by drawing a diagram indicating the movement of missionaries and cannibals.
[Hint/Spoiler: I solved it with 11 moves. A key step is to get to get the left shore (side 0) to MMMCCC i.e., (3,3) with a boat. Then, you send all of the missionaries across and then only work with cannibals to bring the rest across.]

using (M,C) notation

Side ∅		side 1
(5,5)	1,1	(0,0)
(4,4)	1,0	(1,1)
(5,4)	0,1	(0,1)
(5,1)	0,1	(0,4)
(5,2)	2,0	(0,3)
2,2	1,1	(3,3)
(3,3)	3,0	(2,2)
(0,3)	0,1	(5,2)
(0,4)	0,3	(5,1)
(0,1)	0,1	(5,4)
(0,2)	0,2	(5,3)
		(5,5)

I've redone this a few times keep catching mistakes... hopefully it's right



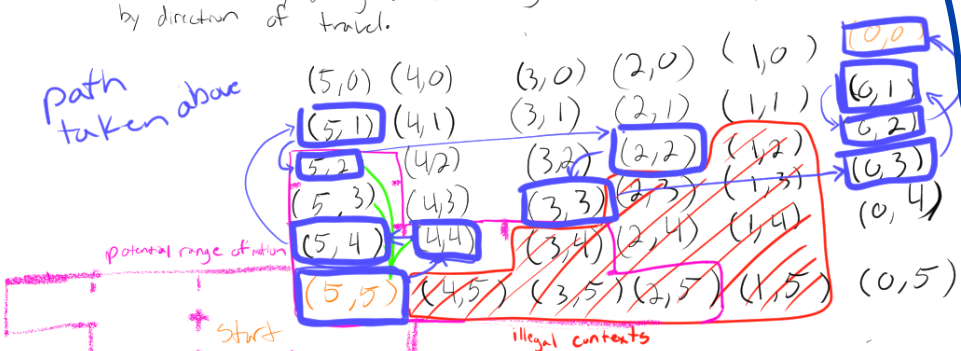
b. Represent the solution to the riddle as a state diagram. (Denote how you are encoding the information.)

confused on this. in context of this class gaugling state diagram turns up this: ↗ which looks like something from my Obj. oriented design course. I have a hunch this isn't what you want?

encoding: side ∅ is represented & stored as a vector, {M,C}, where M is the number of missionaries and C is the number of cannibals. Movement across the river can be described by adding or subtracting a similar vector operation determined by direction of travel.

legal moves:

×	±(0,1)	±(0,2) (0,3)	
±(1,0)	±(1,1)	×	×
±(2,0)	±(2,1)	×	×
±(3,0)	×	×	×



More about this in a bit ...

(4,4) ...

(5,3)

Problem 2 (Bonus). Attempts at implementing a search algorithm (in a language of your choice) to construct a path from initial state to end state in the Farmer River Crossing Riddle or Missionaries and Cannibals Riddle?

rules:
- vectors can only contain whole numbers
- unless M=0 it must be true that C≤M for any vector
- the sum of values in the boat vector cannot exceed 3
- each step alternates adding & subtracting the boat vector from the side ∅ vector, beginning with subtraction

Problem 3. On one bank of a river are **N** missionaries and **N** cannibals. There is one boat available that can hold up to **four** people and that they would like to use to cross the river. If the cannibals ever outnumber the missionaries on either of the river's banks, the missionaries will get eaten. How can the boat be used to safely carry all the missionaries and cannibals across the river?

a. If $N = 2$, how can the boat be used to safely carry all the missionaries and cannibals across the river? (Solve in 1 move)

(3,3)→(2,2)→(3,2)→(3,0)→(3,1)→(1,1)→(2,2)→(0,2)→(0,3)→(0,1)→(1,1)→(0,0)

using this notation ↑ where (M, C) represents side ∅

$(2,2) \rightarrow (0,0)$

b. If $N = 3$, how can the boat be used to safely carry all the missionaries and cannibals across the river? (Solve in 3 moves)

$(3,3) \rightarrow (0,2) \rightarrow (0,3) \rightarrow (0,0)$

c. If $N = 4$, how can the boat be used to safely carry all the missionaries and cannibals across the river? (Solve in 5 moves)

1 2 3 4 5

$(4,4) \rightarrow (2,2) \rightarrow (4,2) \rightarrow (0,2) \rightarrow (0,3) \rightarrow (0,0)$

d. If $N = 5$, how can the boat be used to safely carry all the missionaries and cannibals across the river? (Solve in 7 moves)

$(5,5) \rightarrow (3,3) \rightarrow (4,4) \rightarrow (2,2) \rightarrow (3,3) \rightarrow (1,1) \rightarrow (2,2) \rightarrow (0,0)$

e. (Bonus) Can you say anything in general? Are there any patterns/strategies that emerge?

can run MMCC → MC over and over... :)

if the boat can carry an even number of people ≥ 4, then you can just have a pair of missionary + cannibal representatives ferry the others across. if it is odd & ≥ 5, you can do the same. boats fitting ≤ 3 people will fail eventually if N is too large. :)