Team Activity

created by Jemuel De Guzman and Kurt Russel A. Villamor

Algorithm

- 1 START
- 2 put the sheep from the farm to the boat
- 3 Drop sheep in the other side
- 4 go back to farm
- 6 put cabbage to boat
- 7 swap sheep to cabbage
- 8 go back to farm
- 9 swap sheep to wolf
- 10 drop the wolf to other side
- 11 go back to farm
- 12 put the sheep from the farm to the boat
- 13 Drop sheep in the other side
- 14 END

Classes/Code

```
# Classes
class Cabbage:
    def __init__(self):
        self.name = "Cabbage"

class Sheep:
    def __init__(self):
        self.name = "Sheep"

class Wolf:
    def __init__(self):
        self.name = "Wolf"

class Boat:
    def __init__(self):
        self.passenger = None


    def takePassenger(self, location, passenger):
        if passenger in location:
            location.remove(passenger)
            self.passenger = passenger
            print(f"Taken {self.passenger.name} as a passenger")
            self.checkBoat(location)
        else:
            print(f"No {passenger.name} is on this side")

    def dropPassenger(self, location):
        location.append(self.passenger)
        print(f"Dropped {self.passenger.name}")
        self.checkBoat(location)

    def swap(self, location):
        if len(location) == 1:
            self.alonePassenger = location[0]
            location[0] = self.passenger
            print(f"Swap {self.alonePassenger.name} and {self.passenger.name}")
            self.passenger = self.alonePassenger
            self.checkBoat(location)

    def checkBoat(self, location):
        if len(location) == 2:
            names = []
            for i in location:
                names.append(i.name)
            if "Sheep" in names:
                if "Wolf" in names:
                    print("Wolf is not allowed with the Sheep")
                if "Cabbage" in names:
                    print("Cabbage is not allowed with the Sheep")
```

```
def print_by_loc(location):
    passengers = []
    for i in location:
        passengers.append(i.name)
    print(f"{passengers}")
```

Initialization

1 START

- calling the classes and putting them to a list/array representing the location

```
cabbage = Cabbage()
wolf = Wolf()
sheep = Sheep()
boat = Boat()

Farm = [cabbage, wolf, sheep] # The starting Point of the Problem
OtherSide = [] # The end point
```

2 Put the sheep from the farm to the boat

- From Farm, we get the sheep to the boat.

```
boat.takePassenger(Farm, sheep)
```

```
    Taken Sheep as a passenger
```

3 Drop sheep in the other side

- Drop the sheep to the OtherSide.

```
boat.dropPassenger(OtherSide)
```

```
    Dropped Sheep
```

4 go back to farm 6 put cabbage to boat

- From the Farm, we get the cabbage

```
boat.takePassenger(Farm, cabbage)
```

```
    Taken Cabbage as a passenger
```

7 swap sheep to cabbage

- Since, sheep and cabbage can't be together resulting in an error prompt, we propose that the method dropPassenger() will not be used.

boat.dropPassenger(OtherSide) Output = Dropped Cabbage is not allowed with the Sheep

- as an alternative, we made a swap function for the passenger and alone passenger in the location.

```
boat.swap(OtherSide)
```

```
    Swap Sheep and Cabbage
```

8 go back to farm 9 swap sheep to wolf

- same in these steps, swap is the option since Wolf and Sheep is not allowed together.

```
boat.swap(Farm)
```

```
    Swap Wolf and Sheep
```

10 drop the wolf to other side

- wolf and cabbage has no constrains

```
boat.dropPassenger(OtherSide)
```

```
    Dropped Wolf
```

11 go back to farm 12 put the sheep from the farm to the boat

- same as the first step

```
boat.takePassenger(Farm, sheep)
```

```
    Taken Sheep as a passenger
```

13 Drop sheep in the other side

- same step in the second step

```
```

```
boat.dropPassenger(OtherSide)
```

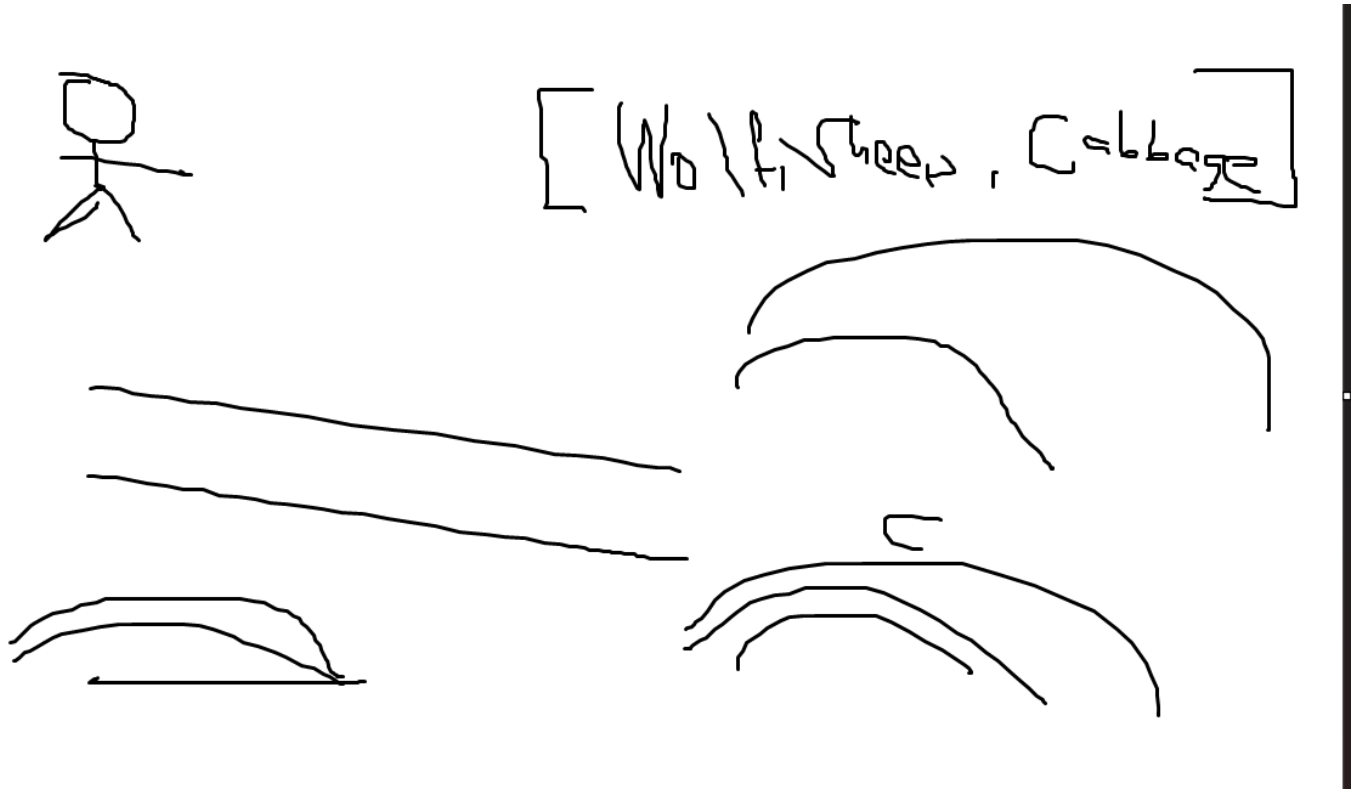
```
    Dropped Sheep
```

14 END

```
print_by_loc(Farm) # Farm Ending State
print_by_loc(OtherSide) # OtherSide Ending State
```

```
    []
    ['Cabbage', 'Wolf', 'Sheep']
```

Representation



```
# Created by Kynamittens and xeno...
```