Assignment 1.2 Exercises on Computational Thinking with Python

Course Code: CPE311

Course Name: Computational Thinking with Python

Section: CPE22S3

Program: BSCPE

Date Started: January 24, 2024

Date Submitted: January 27, 2024

Members:

Galapia, Xander Sam

Garcia, John Carlos

```
class Character:
  def __init__(self, name, side):
   self.name = name
   self.side = side

  def move(self, new_side):
   self.side = new_side

class State:
    def __init__(self, sheep, cabbage, wolf, boat_side):
        self.sheep = sheep
        self.cabbage = cabbage
        self.wolf = wolf
        self.boat_side = boat_side

    def display_current_state(self):
        print("\nCurrent State:")
        print(f"Sheep: {self.sheep.side}")
        print(f"Cabbage: {self.cabbage.side}")
        print(f"Wolf: {self.wolf.side}")
        print(f"Boat: {self.boat_side}\n")

    def is_valid_move(self, character):
        if character.name == "Boat":
            return True
        elif character.name == "Sheep":
            return (

              character.side == self.boat_side or self.boat_is_empty())
        elif character.name == "Cabbage":
            return character.side == self.boat_side or (self.boat_is_empty() and self.sheep.side == character.side)
        else:  # Assume it's the wolf
            return character.side == self.boat_side or (self.boat_is_empty() and self.sheep.side == character.side)

    def boat_has_cabbage(self):
        return self.cabbage.side == self.boat_side

    def boat_is_empty(self):
        return self.sheep.side == self.cabbage.side == self.wolf.side == self.boat_side

    def move_character(self, character):
        if self.is_valid_move(character):
            if character.name == "Boat":
                self.boat_side = "right bank" if self.boat_side == "left bank" else "left bank"
            else:
                character.move("right bank" if character.side == "left bank" else "left bank")
                self.boat_side = "right bank" if self.boat_side == "left bank" else "left bank"
            return True
        else:
            return False
def story():
    print("You're being hunted by an unknown entity and need to go to the other side of the river for it to not reach you.")
    print("\nAs a farmer, you need to transport your sheeps, cabbages, and wolves across the river in order to start anew in a di
    print("\nHowever, you can only take one of them with you in the boat at a time.")
    print("\nBe careful, as leaving the wolf alone with the sheep or the sheep alone with the cabbage will have dire consequences
    print("                                                                  ")
```

```
    print("------------------------------------------------------------")

def main():
  story()

  sheep = Character("Sheep", "left bank")
  cabbage = Character("Cabbage", "left bank")
  wolf = Character("Wolf", "left bank")
  boat = Character("Boat", "left bank")

  game_state = State(sheep, cabbage, wolf, "left bank")

  while True:
    game_state.display_current_state()
    user_input = input("What do you want to move to the other side of the river (s = sheep, c = cabbage, w = wolf, b = boat): ").
    if user_input == "quit":
      print("The void consumes you.")
      break

    character_to_move = None
    if user_input == "s":
      character_to_move = sheep
    elif user_input == "c":
      character_to_move = cabbage
    elif user_input == "w":
      character_to_move = wolf
    elif user_input == "b":
      character_to_move = boat
    else:
      print("Invalid input. Please enter a valid character name or 'quit' to exit.")
      continue

    if game_state.move_character(character_to_move):
      if game_state.sheep.side == "right bank" and game_state.cabbage.side == "right bank" and game_state.wolf.side == "right ban
        print("Congratulations! You successfully transported the sheep, cabbage, and wolf across the river.")
        break
    else:
      print("Invalid move. Please try again.")

if __name__ == "__main__":
    main()


     Be careful, as leaving the wolf alone with the sheep or the sheep alone with the cabbage will have dire consequences!

     ------------------------------------------------------------

     Current State:
     Sheep: left bank
     Cabbage: left bank
     Wolf: left bank
     Boat: left bank

     What do you want to move to the other side of the river (s = sheep, c = cabbage, w = wolf, b = boat): s

     Current State:
     Sheep: right bank
     Cabbage: left bank
     Wolf: left bank
     Boat: right bank

     What do you want to move to the other side of the river (s = sheep, c = cabbage, w = wolf, b = boat): b

     Current State:
     Sheep: right bank
     Cabbage: left bank
     Wolf: left bank
     Boat: left bank

     What do you want to move to the other side of the river (s = sheep, c = cabbage, w = wolf, b = boat): c

     Current State:
```

```
Current State:
Sheep: left bank
Cabbage: right bank
Wolf: left bank
Boat: left bank

What do you want to move to the other side of the river (s = sheep, c = cabbage, w = wolf, b = boat): w

Current State:
Sheep: left bank
Cabbage: right bank
Wolf: right bank
Boat: right bank

What do you want to move to the other side of the river (s = sheep, c = cabbage, w = wolf, b = boat): b

Current State:
Sheep: left bank
Cabbage: right bank
Wolf: right bank
Boat: left bank
```