**Technological Institute of the Philippines**

Computer Engineering Department

Quezon city Campus

TEAM CHALLENGE

| Course: CPE 311 | Program: BSCpE |
|---|---|
| **Course Title**: Computional Thinking in Python | **Date Performed:**1/24/2024 |
| **Section**:CPE22S3 | **Date Submitted:** 1/24/2024 |
| **Student Name**: Jhillian Cabos, Christian Lei Calingo | **Instructor's Name:** Roman Richard |

```
from google.colab import files
from IPython.display import Image
uploaded = files.upload()
```

Choose Files No file chosen Upload widget is only available when the cell has been executed in the current browser session. Please rerun this cell to enable.

Saving happyending.jpg to happyending (1).jpg

+ Code + Text

```
from IPython.display import Image, display
class Place :
  def __init__(self):
    #declaring the list of the characters starting position
    self.leftside = ["Wolf", "Sheep", "Cabbage", "Jack"]
    self.rightside = []
class Eating:
    def __init__(self, place):
      #a trigger if by some chance the user enters the wrong answer
      self.life = 1
      #accessing the leftside and the rightside of the Place class
      self.leftside = place.leftside; self.rightside = place.rightside

    def eating(self):
      #creating a function for checking the status of each sides
      self.check_status(self.leftside); self.check_status(self.rightside)

    def check_status(self, side):
      #checking if the wolf and sheep is on the same side but jack is not
      if "Wolf" in side and "Sheep" in side and "Jack" not in side:
          print("The Sheep was Eaten")
          self.life -= 1
      #checking if the wolf and sheep is on the same side but jack is with them
      elif "Wolf" in side and "Sheep" in side and "Jack" in side:
          pass
      #checking if the cabbage and sheep is on the same side but jack is not
      if "Cabbage" in side and "Sheep" in side and "Jack" not in side:
          print("The Cabbage was Eaten")
          self.life -= 1
      #checking if the cabbage and sheep is on the same side but jack is with them
      elif "Cabbage" in side and "Sheep" in side and "Jack" in side:
          pass

    def switchside_left(self, char):
      #moving jack to the leftside and the transported passenger
      self.leftside.remove(char); self.leftside.remove("Jack")
      self.rightside.append("Jack"); self.rightside.append(char)

    def switchside_right(self, char):
      #moving jack to the rightside and the transported passenger
      self.rightside.remove(char); self.rightside.remove("Jack")
      self.leftside.append("Jack"); self.leftside.append(char)

    def print_sides(self):
      #printing the status of each sides
      print("-> On the rightside: ", ', '.join(map(str, self.rightside)))
      print("-> On the leftside: ", ', '.join(map(str, self.leftside)))
class Prompt:
    def __init__(self, eat, left, right, switch_left, switch_right):
      #User propmt
      print("Jack has to transport Wolf, Cabbage, and Sheep, He can only carry one passenger per trip")
      answer = input("Who is coming with Jack?\n-> ")
      while answer not in left:
```

```
        print("No such thing")
        answer = input("Who is coming with Jack?\n-> ")
      switch_left(answer); eat.eating()
      if eat.life == 0: print("Oh no")
      else:
        print("------------------\nJack reached the right side with the", answer)
        eat.print_sides()
        print("------------------\nAfter transporting the " + answer + ", Jack goes back to the left side")
        right.remove("Jack"); left.append("Jack")
        eat.print_sides()
        print("------------------")
        answer2 = input("Jack is going back to the right side, who is coming with Jack?\n-> ")
        while answer2 not in left:
          print("No such thing")
          answer2 = input("Who is coming with Jack?\n-> ")
        print("------------------\nJack reached the right side with the", answer2)
        switch_left(answer2)
        eat.print_sides()
        print("------------------\nEnter None if no one")
        answer3 = input("Jack is going back to the left side, who is coming with Jack?\n-> ")
        if answer3 == "None":
          right.remove("Jack")
          left.append("Jack")
          print("Oh no")
          eat.eating()
        else:
          switch_right(answer3)
          eat.eating()
          print("------------------\nJack reached the left side with the ", answer2)
          eat.print_sides()
          print("------------------")
          print("Enter None if no one")
          answer4 = input("Jack is going back to the right side, who is coming with Jack?\n-> ")
          if answer4 == "None":
            left.remove("Jack"); right.append("Jack")
            print("Oh no")
            eat.eating()
          else:
            switch_left(answer4)
            eat.eating()
            eat.print_sides()
            print("------------------\nSince the Wolf won't  eat the cabbage, Jack decides to go alone in the left side\n------------------\n
            right.remove("Jack"); left.append("Jack"); eat.print_sides()
            print("------------------")
            answer5 = input("Jack is going back to the right side, who is coming with Jack? ")
            while answer5 not in left:
              print("No such thing")
              answer5 = input("Who is coming with Jack? ")
            switch_left(answer5); eat.eating()
            print("------------------\nJack reached the right side with the "+answer5+"\n------------------");eat.print_sides()
            if len(right) == 4: print("Jack Have Successfully transported all of them");display(Image(filename='happyending.jpg', width='500'
eat = Eating(place=Place())
left, right = eat.leftside, eat.rightside
switch_left, switch_right = eat.switchside_left, eat.switchside_right
Pr = Prompt(eat, left, right, switch_left, switch_right)
```

```
Jack has to transport Wolf, Cabbage, and Sheep, He can only carry one passenger per trip
Who is coming with Jack?
-> Sheep
------------------
Jack reached the right side with the Sheep
-> On the rightside:  Jack, Sheep
-> On the leftside:  Wolf, Cabbage
------------------
After transporting the Sheep, Jack goes back to the left side
-> On the rightside:  Sheep
-> On the leftside:  Wolf, Cabbage, Jack
------------------
Jack is going back to the right side, who is coming with Jack?
-> Wolf
------------------
Jack reached the right side with the Wolf
-> On the rightside:  Sheep, Jack, Wolf
-> On the leftside:  Cabbage
------------------
Enter None if no one
Jack is going back to the left side, who is coming with Jack?
-> Sheep
------------------
Jack reached the left side with the  Wolf
-> On the rightside:  Wolf
-> On the leftside:  Cabbage, Jack, Sheep
------------------
Enter None if no one
Jack is going back to the right side, who is coming with Jack?
-> Cabbage
-> On the rightside:  Wolf, Jack, Cabbage
-> On the leftside:  Sheep
------------------
Since the Wolf won't  eat the cabbage, Jack decides to go alone in the left side
------------------
Jack reached the left side with no one
-> On the rightside:  Wolf, Cabbage
-> On the leftside:  Sheep, Jack
------------------
Jack is going back to the right side, who is coming with Jack? Sheep
------------------
Jack reached the right side with the  Sheep
-> On the rightside:  Wolf, Cabbage, Jack, Sheep
-> On the leftside:
------------------
Jack Have Successfully transported all of them
```