Programming Fundamentals Coursework Report

-Imandi Karunaratne

-IIT Student ID (20200306)

Test Data For Omi Code

Test Case	Test Case Description	Test Data	Expected Result	Result(Pass/Fail)	Implemented On code
When both user and robo enters same suit	Robo inputs Queen of Hearts User inputs 9 of Hearts	Robo puts: ♥Q User puts: ♥9	`if suit1 == suit:  # if user has a suit similar to robo suit` `    if int(number1) > int(number):  # checks if the number on card of user is greater than the number on card of robo`	PASS	YES
If both robo and user enters different suits but user puts a trump card	Robo inputs 7 of Diamonds User inputs King of Hearts (trump suit) Suit1=user suit Suit=robo suit	robo puts: ♦7 user puts: ♥K	`elif suit1 != suit:  # checks if the user card suit is not similar to robo suit` `    if suit1 == trump:  # if user has put a card with the trump suit`	PASS	YES
If both robo and user enters different suits but robo puts a trump card	User inputs 7 of Spades Robo inputs 9 of Hearts (trump suit)	user puts: ♠7 robo puts: ♥9	`elif suit3 != suit:  # if robo doesnt have card with similar suit of robo` `    if suit3 == trump:  # checks if robo has a trump`	PASS	YES
Both user and robo enters different suits but both suits are not trumps	(user inputs the card first) User inputs King of Clubs Robo inputs 9 of Diamonds	user puts: ♣K robo puts: ♦9	`elif suit3 != suit:  # if robo doesnt have card with similar suit of robo` `    if suit3 == trump:  # checks if robo`	PASS	YES

			has a trump robopoints += 2 # robo (computer) wins the trick print("computer wins this trick2") tricklead = "robo" # since robo wins this trick the next trick is lead by robo(computer) else: userpoints += 2 # if robo has not put a trump nor a card of similar suit print("you win this trick2") tricklead = "user" # since user wins this trick the next trick is lead by you(user)		
Both user and robo enters different suits but both suits are not trumps	(Robo inputs the card first) Robo inputs Jack of Diamonds User inputs King of Spades	Robo puts: ♦J user puts: ♠K	elif suit1 != suit: # checks if the user card suit is not similar to robo suit if suit1 == trump: # if user has put a card with the trump suit userpoints += 2 # user wins the trick print("you win this trick10") tricklead = "user"	PASS	YES

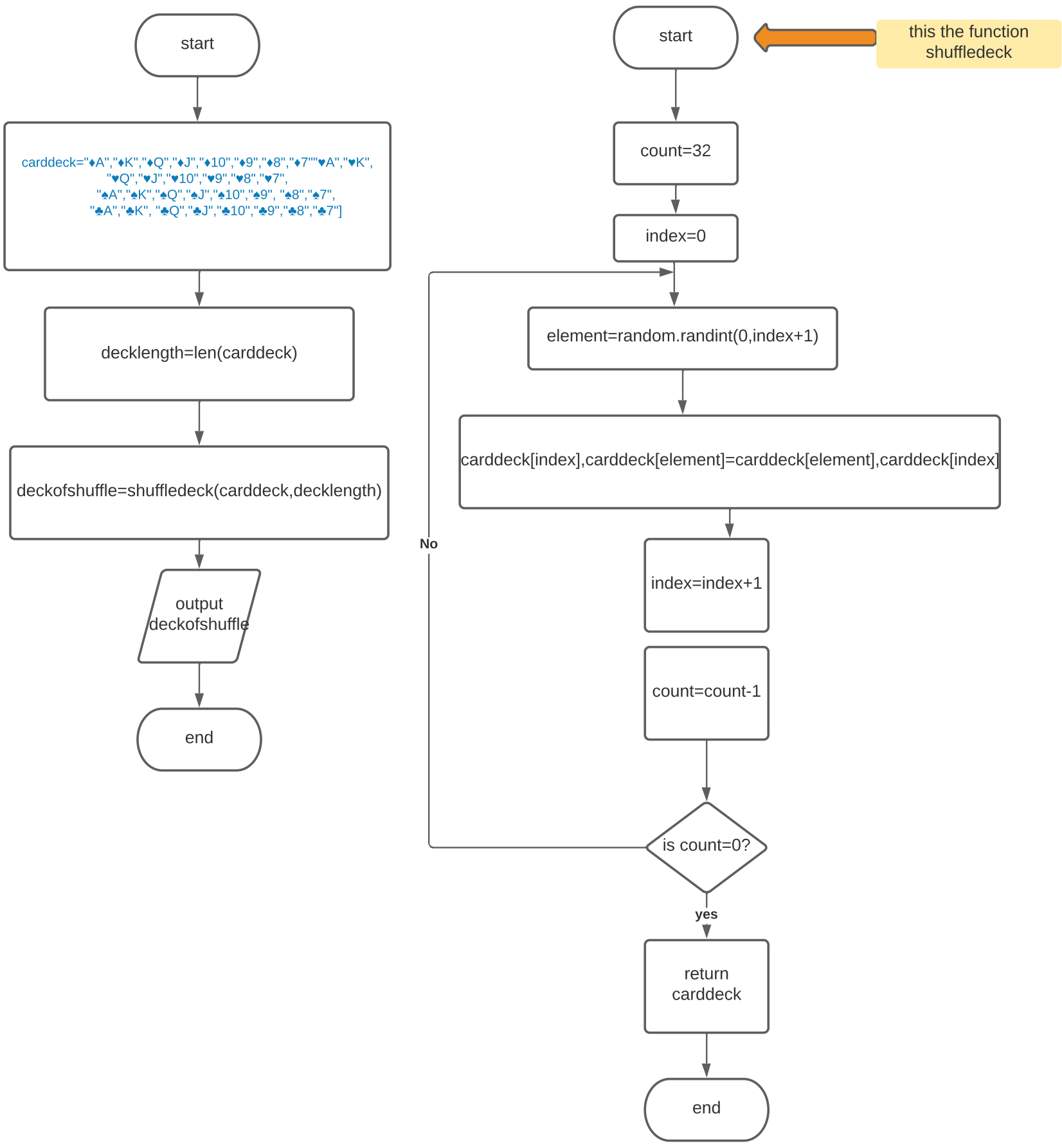
			else: robopoints += 2 # if robo has not put a trump nor a card of similar suit print("computer wins this trick11") tricklead = "robo"		

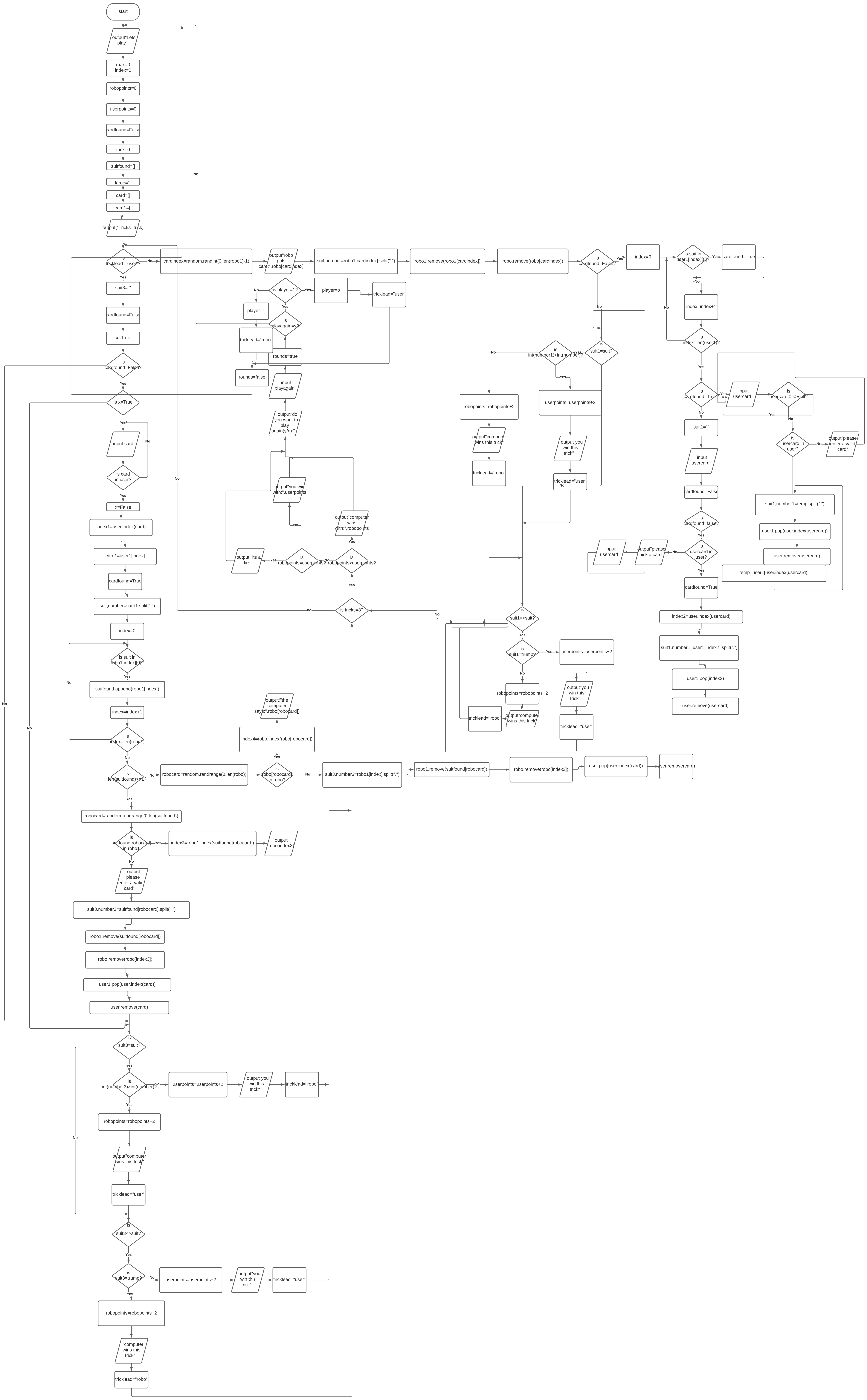
```
carddeck="♦A","♦K","♦Q","♦J","♦10","♦9","♦8","♦7""♥A","♥K",
"♥Q","♥J","♥10","♥9","♥8","♥7",
"♠A","♠K","♠Q","♠J","♠10","♠9", "♠8","♠7",
"♣A","♣K", "♣Q","♣J","♣10","♣9","♣8","♣7"]

decklength=len(carddeck)

def shuffledeck(carddeck,decklength):
    for index in range(decklength-1,0,-1):
        element=random.randint(0,index+1)
        carddeck[index], carddeck[element] =
    carddeck[element], carddeck[index]

    return carddeck
```




start
output"Lets play"
max=0
index=0
robopoints=0
userpoints=0
cardfound=False
trick=0
suitfound=[]
large=""
card=[]
card1=[]
output("Tricks",trick)
is tricklead="user"?
No
Yes
cardindex=random.randint(0,len(robo1)-1)
output"robo puts card:",robo[cardindex]
suit,number=robo1[cardindex].split(".")
robo1.remove(robo1[cardindex])
robo.remove(robo[cardindex])
is cardfound=False?
Yes
index=0
is suit in user1[index][0]?
Yes
cardfound=True
No
index=index+1
is index=len(user1)?
No
Yes
is cardfound=True?
Yes
input usercard
is usercard[0]<>suit?
Yes
No
is usercard in user?
No
output"please enter a valid card"
suit1,number1=temp.split(".")
user1.pop(user.index(usercard))
user.remove(usercard)
temp=user1[user.index(usercard)]
No
suit1=""
input usercard
cardfound=False
is cardfound=false?
Yes
is usercard in user?
No
output"please pick a card"
input usercard
Yes
cardfound=True
index2=user.index(usercard)
suit1,number1=user1[index2].split(".")
user1.pop(index2)
user.remove(usercard)
No
is suit1=suit?
Yes
is int(number1)>int(number)?
No
Yes
robopoints=robopoints+2
output"computer wins this trick"
tricklead="robo"
userpoints=userpoints+2
output"you win this trick"
tricklead="user"
No
is suit1<>suit?
No
Yes
is suit1=trump?
Yes
userpoints=userpoints+2
output"you win this trick"
tricklead="user"
No
robopoints=robopoints+2
output"computer wins this trick"
tricklead="robo"
is tricks=8?
no
Yes
is robopoints>userpoints?
Yes
output"computer wins with:",robopoints
No
is robopoints=userpoints?
Yes
output "its a tie"
No
output"you win with:",userpoints
output"do you want to play again(y/n):"
input playagain
rounds=true
rounds=false
is playagain=y?
No
Yes
is player=1?
Yes
player=o
tricklead="user"
No
player=1
tricklead="robo"
suit3=""
cardfound=False
x=True
is cardfound=False?
No
Yes
is x=True
No
Yes
input card
is card in user?
No
Yes
x=False
index1=user.index(card)
card1=user1[index]
cardfound=True
suit,number=card1.split(".")
index=0
is suit in robo1[index][0]?
Yes
suitfound.append(robo1[index])
index=index+1
is index=len(robo1)
No
No
is len(suitfound)>=1?
No
robocard=random.randrange(0,len(robo))
is robo[robocard] in robo?
Yes
index4=robo.index(robo[robocard])
output("the computer says:",robo[robocard])
No
suit3,number3=robo1[index].split(".")
robo1.remove(suitfound[robocard])
robo.remove(robo[index3])
user.pop(user.index(card))
user.remove(card)
Yes
robocard=random.randrange(0,len(suitfound))
is suitfound[robocard] in robo1
Yes
index3=robo1.index(suitfound[robocard])
output robo[index3]
No
output "please enter a valid card"
suit3,number3=suitfound[robocard].split(".")
robo1.remove(suitfound[robocard])
robo.remove(robo[index3])
user1.pop(user.index(card))
user.remove(card)
is suit3=suit?
yes
is int(number3)>int(number)?
No
userpoints=userpoints+2
output"you win this trick"
tricklead="robo"
Yes
robopoints=robopoints+2
output"computer wins this trick"
tricklead="user"
No
is suit3<>suit?
Yes
is suit3=trump?
No
userpoints=userpoints+2
output"you win this trick"
tricklead="user"
Yes
robopoints=robopoints+2
"computer wins this trick"
tricklead="robo"