

Task:-13:- Finding the winning strategy in a card game.

Aim:- To implement a python program that simulates a card game between two players, and determine the winning strategy using of drawn cards.

Algorithm:-

1. Start the program.
2. Create a deck of cards.
3. Shuffle the deck randomly.
4. Each player, draws a fixed number of cards.
5. Define the strategy.
6. Compare the chosen cards of both players.
  - If Player 1's card > Player 2's card → Player 1
  - Else if Player 2's card > Player 1's card → Player 2
  - Else → Draw.

Scores.

7. Repeat for all rounds.
8. The player with the highest score wins the game.
9. Display input, output, and final result
10. End the program.

Program :-

```
import random  
suits = ["Hearts", "Diamonds", "Clubs", "Spades"]  
values = list(range(1, 14))
```

deck = [(value, suit) for suit in suits for value in values].

```
random.shuffle(deck)
```

Player 1 - hand = deck[:5]

Player 2 - hand = deck[5:10]

```
print("Player 1 Hand:", Player 1 - hand)
```

```
print("Player 2 Hand:", Player 2 - hand).
```

def Play\_highest\_card(hand):

highest = max(hand, key=lambda X: X[1]).

hand.remove(highest)

return highest.

P1 - score, P2 - score = 0, 0

```
print("In -- Game Rounds --")
```

for i in range(5):

P1 - card = play\_highest\_card(Player 1 - hand)

P2 - card = play\_highest\_card(Player 2 - hand).

Print(f"Round {i+1}: Player 1 → {P1 - card},

Player 2 → {P2 - card})

if P1 - card > P2 - card:

```
print("Winner: player 1"),
```

P1 - score += 1

Sample input:-

Player 1 Hand:  $\{ (1, \text{'Hearts'}), (2, \text{'Clubs'}), (10, \text{'Spades'}), (5, \text{'Diamonds'}), (7, \text{'Clubs}) \}$ .

Player 2 Hand:  $\{ (9, \text{'Hearts'}), (12, \text{'Diamonds'}), (3, \text{'Clubs'}), (11, \text{'Spades'}), (6, \text{'Hearts'}) \}$ .

Sample output:-

Round 1: Player 1  $\rightarrow (13, \text{'Hearts'})$ , Player 2  $\rightarrow (12, \text{'Diamonds'})$ .

Winner: Player 1.

Round 2: Player 2  $\rightarrow (10, \text{'Spades'})$ , Player 2  $\rightarrow (11, \text{'Spades'})$ .

Winner: Player 2.

Round 3: - Player 1  $\rightarrow (7, \text{'Clubs'})$  Player 2  $\rightarrow (9, \text{'Hearts'})$ .

Winner: Player 2.

Round 4: Player 1  $\rightarrow (5, \text{'Diamonds'})$ , Player 2  $\rightarrow (6, \text{'Hearts'})$ .

Winner: Player 2

Round 5: Player 1  $\rightarrow (2, \text{'Clubs'})$ , Player 2  $\rightarrow (3, \text{'Clubs'})$ .

Winner: Player 2.

~~Old & New States~~

```

        elif P2 - card[0] > P1 - card[0]:
            Print("winner: player 2").
            P2 - score += 1,
        else:
            Print("Result: draw")

        Print("\n---final Result ---")
        Print("Player 1 score:", P1 - score),
        Print("Player 2 score:", P2 - score)

    if P1 - score > P2 - score:
        Print("Player 1 wins the game with winning
              strategy !").

    elif P2 - score > P1 - score:
        Print("Player 2 wins the game with winning
              strategy !")

    else:
        Print("the game is a draw!").

```

VEL TECH - CSE	
EX NO.	13
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	5
CORD (5)	13
DATE	13/10/2015
TH DATE	13/10/2015

Rony  
is 10/10 ✓

Result: - Thus, the finding winning strategy in a card game is executed successfully.