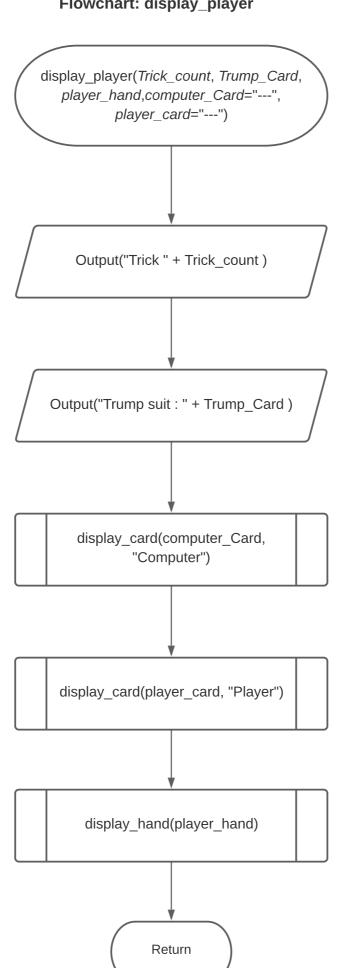
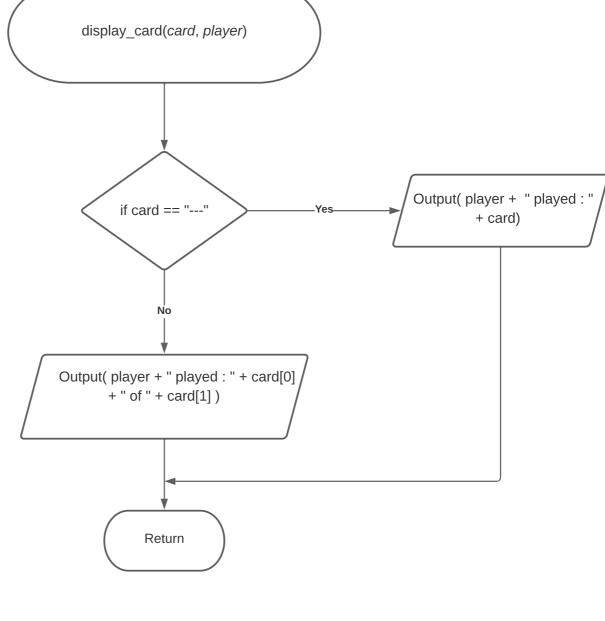


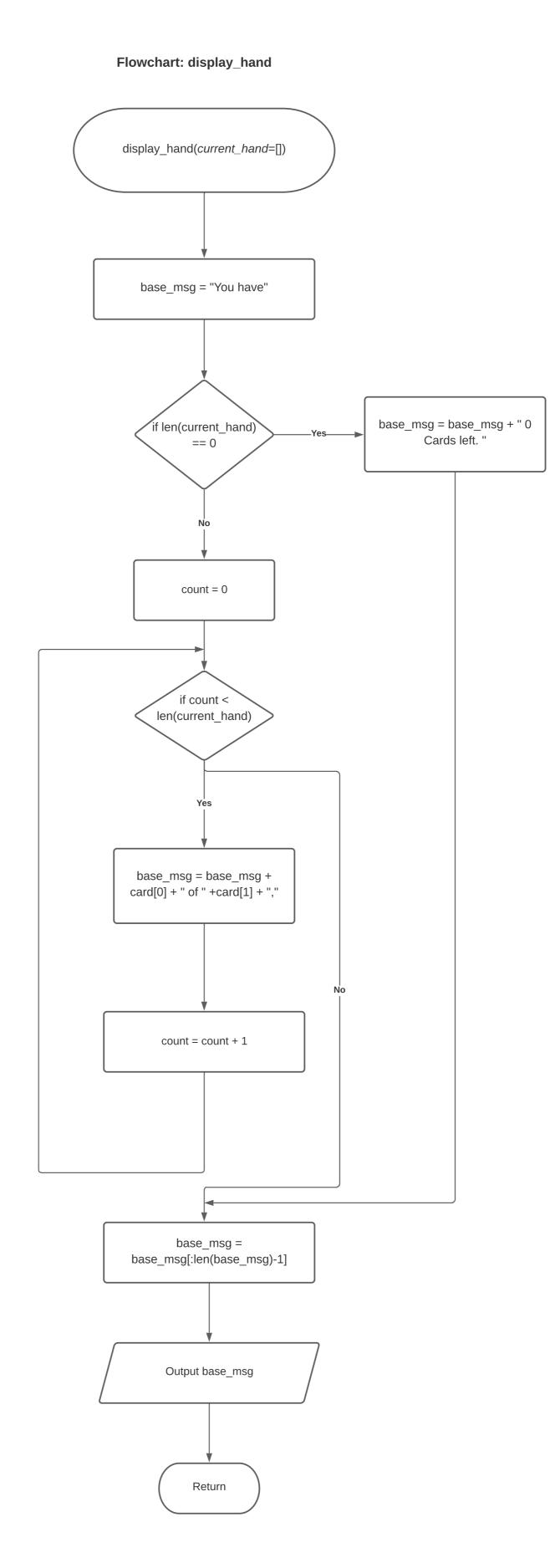
Return winner

Flowchart: display_player



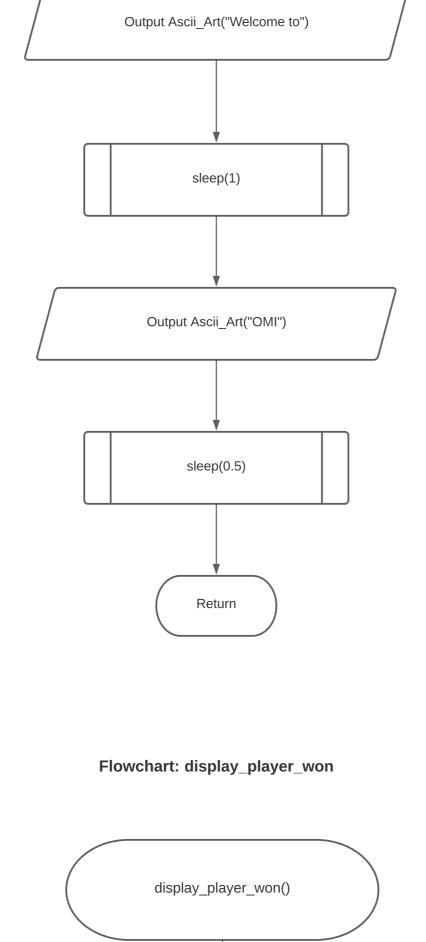
Flowchart: display_card





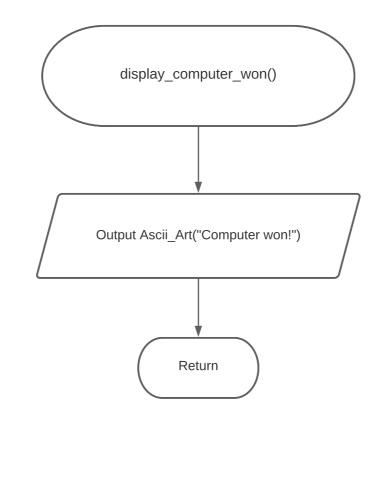
Flowchart: display_welcome_msg

display_welcome_msg()



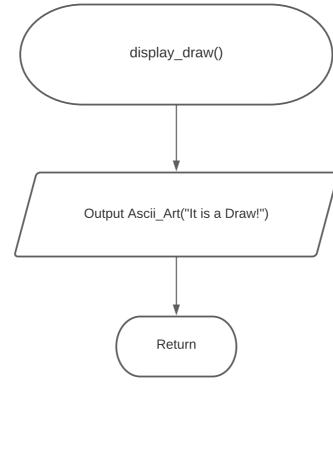
Return

Output Ascii_Art("You won!")

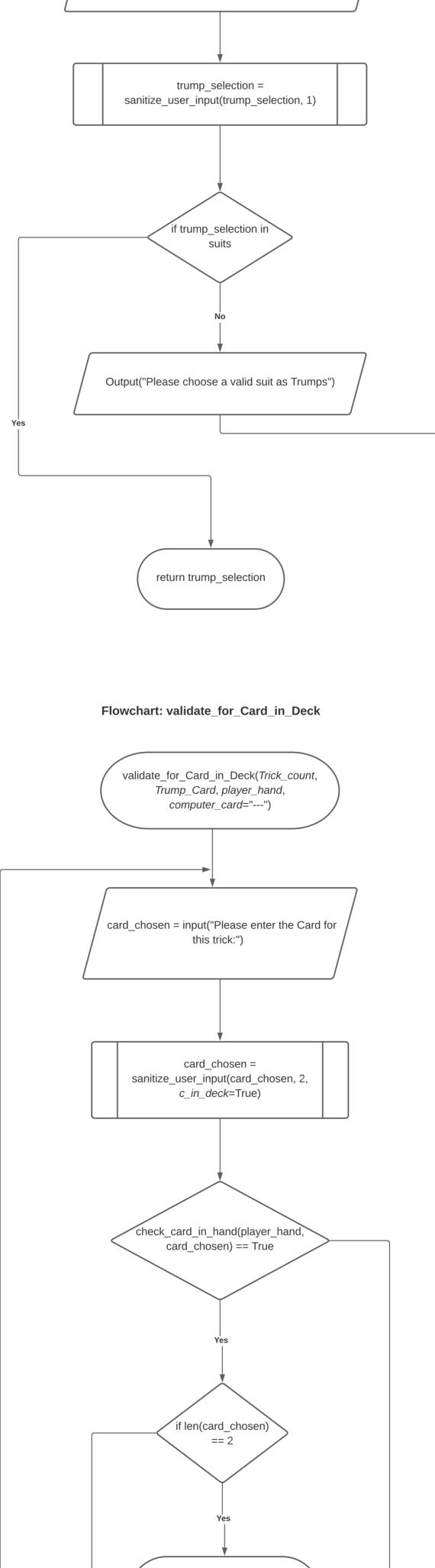


Flowchart: display_computer_won

Flowchart: display_draw



Below are the Validation functions used. Flowchart: validate_trick_play $validate_trick_play(computer_Card,$ player_card, player_deck) if computer_Card[1] != player_card[1] suit = computer_Card[1] $card_With_suit = [c for c in player_deck if c[1]]$ == suit] if len(card_With_suit) > 0 Yes return False Νο return True Flowchart: sanitize_user_input sanitize_user_input(card, upper_bound=2, **kwargs) $card = re.sub(r'\s^*', ", card)$ card = card.capitalize() sanitized_card = card if len(card) > upper_bound Yes sanitized_card = card[:upper_bound] if(kwargs and kwargs["c_in_deck"] == True) No if(card[1] == '0') Νo Yes sanitized_card = Νο card[:upper_bound+1] return sanitized_card Flowchart: check_card_in_hand check_card_in_hand(current_hand, card) $in_{ext} = any((card == x[0]+x[1] for x in$ current_hand)) return in_deck Flowchart: validate_trump_suit validate_trump_suit() suits = ("♠", "♠", "♥", "♦") trump_selection = Input("Please enter the trump suit:") trump_selection = sanitize_user_input(trump_selection, 1) if trump_selection in suits Output("Please choose a valid suit as Trumps") Yes return trump_selection Flowchart: validate_for_Card_in_Deck



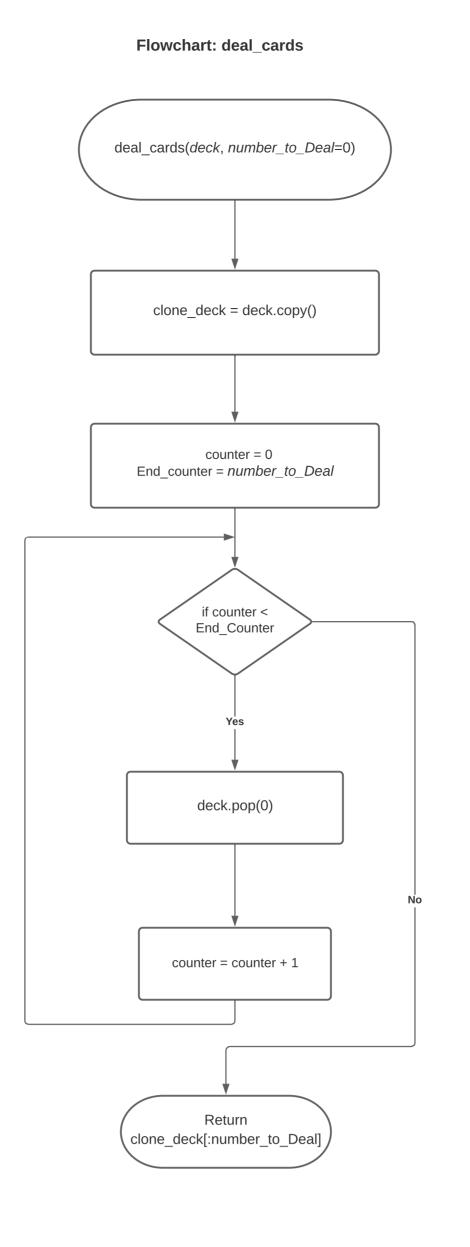
return (card_chosen[0], card_chosen[1])

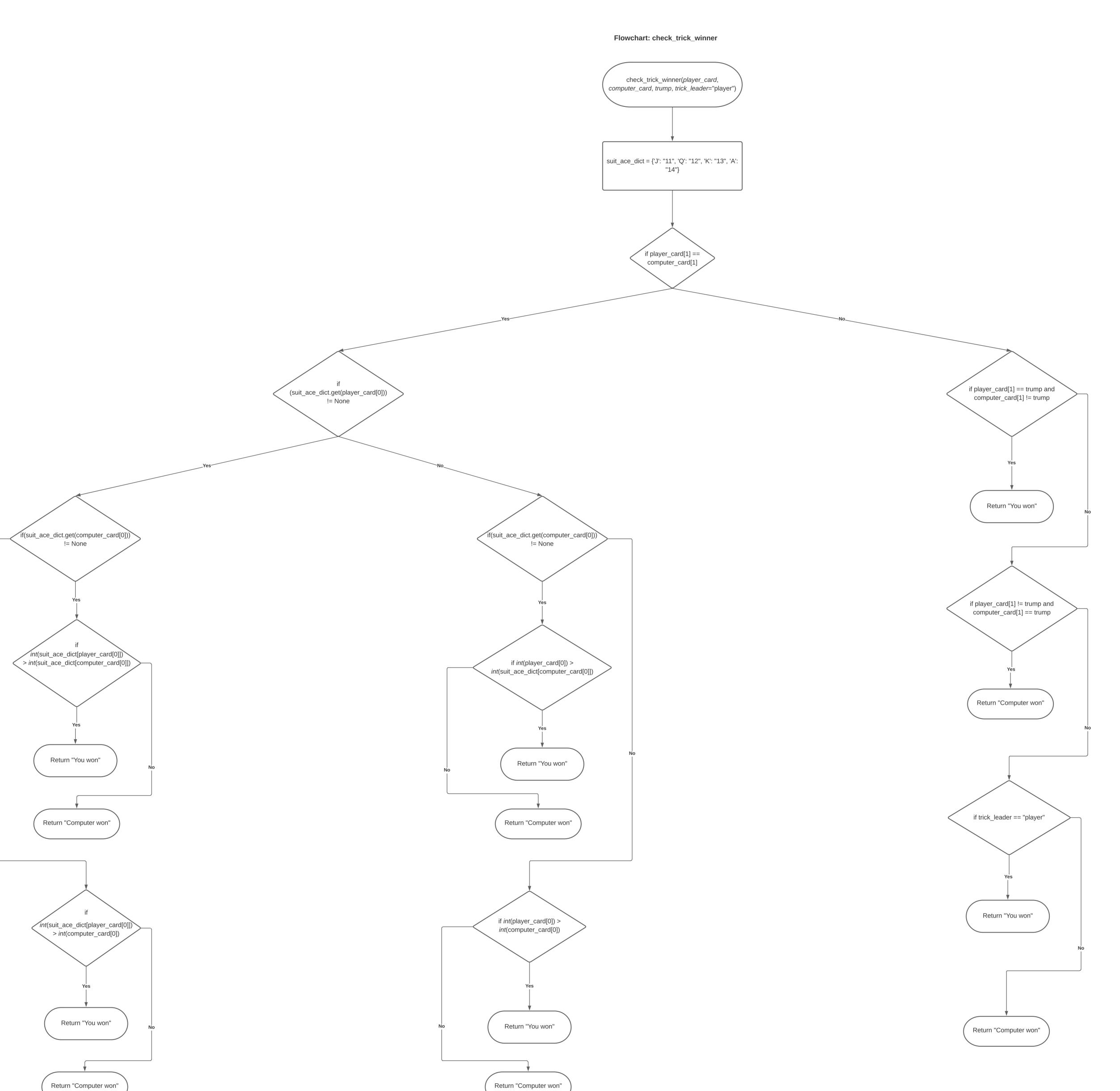
return
(card_chosen[0:len(card_chosen)-1],
card_chosen[-1])

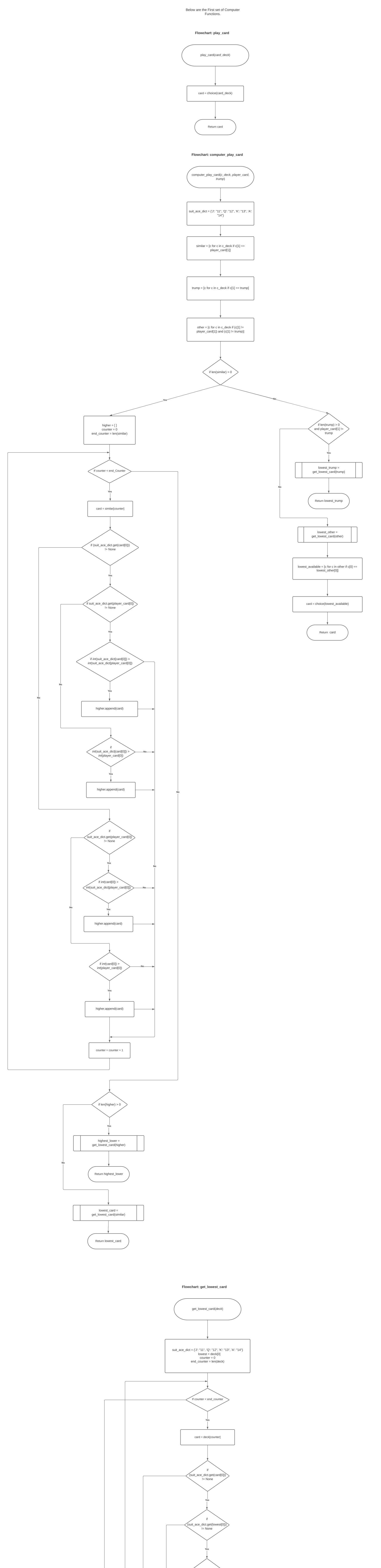
print("Please choose a valid card in your deck!")

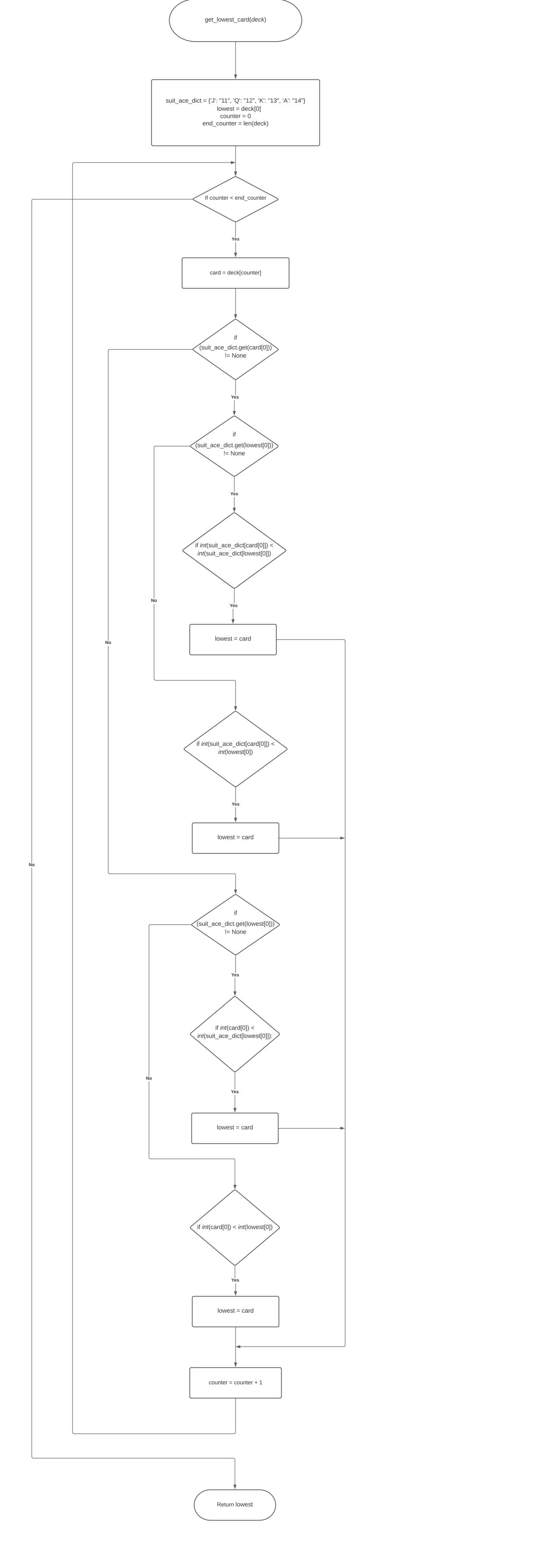
display_player(Trick_count, Trump_Card, player_hand, computer_card) No

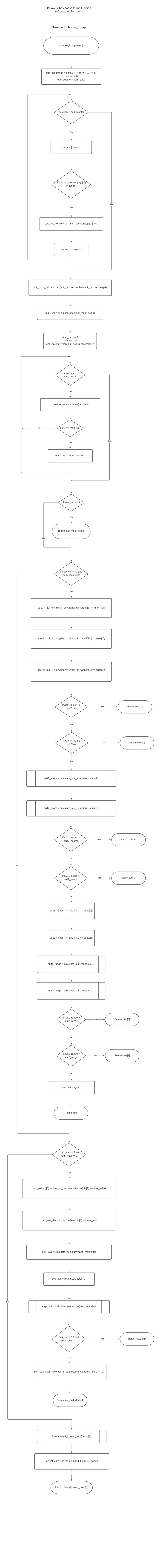
No



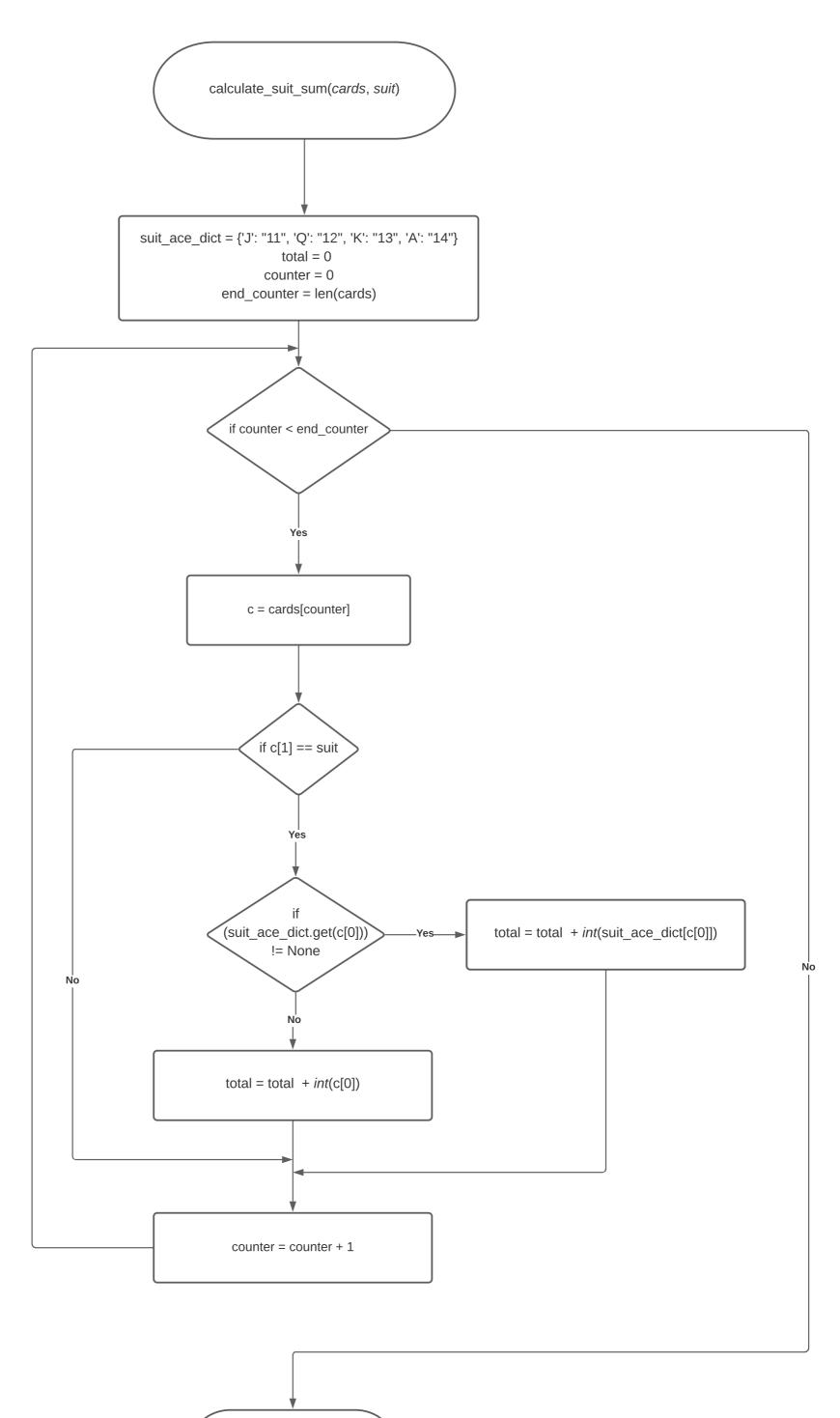








Flowchart: calculate_suit_sum



Flowchart: calculate_suit_range

Return total

