NAME	PURPOSE	TYPE	LIMITATIONS
NUM_VALUES	Holds the number of values	Integer	13
NUM_SUITS	Holds the number of suits	Integer	4
NUM_CARDS	Holds the total number of cards in a deck	Integer	52
CARD_DECK	Determines whether a card is located in the deck or not	2D 4*13 boolean list, row index is card suit and column index is card value	True / False
CARD_VALUE	Allows a column index from CARD_DECK to determine the value of a card	1D string list of card values (typed out)	[‘ACE’, ‘TWO’, …, ‘KING’, ‘ACE’]
CARD_TYPE	Allows a row index from CARD_DECK to determine the suit of a card	1D string list of card types	[‘CLUB’, ‘DIAMOND’, ‘HEART’, ‘SPADE’]
PLAYER_ONE_BAL	Holds the number of chips that player one currently has	Integer	>0
PLAYER_TWO_BAL	Holds the number of chips that player two currently has	Integer	>0
BET	Holds the bet placed by the current player	Integer	0 <= BET <= player balance or pot balance, whichever is smaller
POT	Holds the current number of chips in the pot	Integer	>=0
CARDS_DRAWN	Holds the number of cards that have already been drawn from the deck	Integer	<=52
TURN	Determines whose turn it currently is. TURN = True if it is player one’s turn, and TURN = False if it is player two’s turn	Boolean	True/False
GAME_RUNNING	Creates an infinite loop for the GAME() function. The loop is exited when the game ends using “return”, as making GAME_RUNNING =	Boolean	True

	False would still allow the loop to complete before exiting.		
CARD_ONE	Holds the value and suit of the first card drawn	String	[‘0->12’, ‘0->3’]
CARD_TWO	Holds the value and suit of the second card drawn	String	[‘0->12’, ‘0->3’]
CARD_THREE	Holds the value and suit of the third card drawn	String	[‘0->12’, ‘0->3’]
ANTE	Holds the value of the ante	Integer	1

NAME	PURPOSE	RECEIVES	RETURNS
INIT_DECK()	Initializes the deck	Nothing	A Boolean 2D 4*13 array of True values
GET_CARD()	Finds NUM_DRAWN random cards in a card deck, reshuffles the deck if there are no cards left	NUM_DRAWN CARDS_DRAWN CARD_DECK	CARDS_DRAWN (changes based on how many cards were requested) CARD_RETURN (1D array with each index being one of the cards that was drawn) CARD_DECK (changes values of the array to False if the card in that index position was drawn)
COLLECT_ANTE()	Automatically collects the ante from both players if the pot is empty, and prints player balances regardless.	PLAYER_ONE_BAL PLAYER_TWO_BAL POT	PLAYER_ONE_BAL (subtracts if ante is collected) PLAYER_TWO_BAL (subtracts if ante is collected) POT (adds if ante is collected)
QUIT()	Prints the results of the game when called (final balances and who won)	PLAYER_ONE_BAL PLAYER_TWO_BAL	Nothing

CHECK_ACE()	If any of the cards are aces, ask the player if they would like to make it high or low. If both cards are aces, automatically makes one high and one low.	TURN (to determine who to address when asking the question) CARD_ONE CARD_TWO	CARD_ONE (changes if player makes ace high/low) CARD_TWO (changes if player makes ace high or low)
DEAL_CARD()	Asks the player if they would like their cards dealt to them or if they would like to quit. If they would like their cards dealt, call GET_CARD() and CHECK_ACE(), otherwise call QUIT()	PLAYER_ONE_BAL (to call QUIT() if player wants to quit) PLAYER_TWO_BAL (to call QUIT() if player wants to quit) TURN CARDS_DRAWN (because calling GET_CARD()) CARD_DECK (because calling GET_CARD())	CARDS_DRAWN CARD_DECK CARD_ONE CARD_TWO
CHECK_CASE()	Checks the three different cases in the game (cards are same, consecutive, else). If the cards are the same, give player two chips from pot. If the cards are consecutive, the player cannot bet. If the cards are anything else, the player can bet.	CARD_ONE CARD_TWO POT TURN PLAYER_ONE_BAL PLAYER_TWO_BAL	BET PLAYER_ONE_BAL PLAYER_TWO_BAL POT
IN_BETWEEN()	If the case from CHECK_CASE is else and the bet is >0, this function is called. Draws the third card and checks if this card is in between the first two. If it is, the bet is returned to the player. If it is not, the bet is added to the pot.	CARD_ONE CARD_TWO POT TURN PLAYER_ONE_BAL PLAYER_TWO_BAL	POT PLAYER_ONE_BAL PLAYER_TWO_BAL
GAME()	Initializes CARD_DECK, PLAYER_ONE_BAL, PLAYER_TWO_BAL, BET, POT, CARDS_DRAWN,	None	None

	GAME_RUNNING, TURN. In a while GAME_RUNNING loop: runs COLLECT_ANTE, checks if either player is out of chips. Runs DEAL_CARD, checks if player requested to quit. Runs CHECK_CASE. Gets a third card by running GET_CARD. Runs IN_BETWEEN, and at the end of the loop flips TURN.		
BUBBLE_SORT()	Uses a modified bubble sort to sort the two cards in ascending order	CARD_ONE CARD_TWO	CARD_ONE CARD_TWO