```
PROGRAM START

INITIALIZE user_wins, cpu_wins to 0
SET play_again to True

WHILE play_again is TRUE
    PROMPT user for their name and store in user_name

    RANDOMLY SELECT current_player as "User" or "CPU"
    PRINT "Game starting! [current_player] goes first."
    PRINT "Generating sticks,get ready!"

    SET sticks to a list containing [1,3,5,7]

    WHILE sum of values in sticks is grater than 0:
        FOR each row in sticks:
            PRINT the row number and the corresponding sticks as a pyramid shape

        IF current_player is "User":
            SET valid_input to False
            WHILE valid_input is False:
                TRY:
                    PROMPT user to enter row number (1-4) and store input in row_index_input
                    IF row_index_input is 'q' (case-insensitive):
                        PRINT "Thanks for playing!"
                        EXIT the program
                    CONVERT row_index_input to integer and store in row_index
                CATCH invalid input:
                    PRINT "Invalid row number. Try again."
                    CONTINUE to the next iteration of the loop

                IF row_index is out of range:
                    PRINT "Invalid row number. Try again."
                    CONTINUE to the next iteration of the loop

                PROMPT user to enter number of sticks to take from the selected row
                CONVERT user input to integer and store in sticks_to_take
                IF sticks_to_take is out of range for the selected row:
                    PRINT error message
                    CONTINUE to the next iteration of the loop
                ELSE:
                    SET valid_input to True

            REMOVE sticks_to_take from the selected row in sticks
            PRINT move made by the user
            SET current_player to "CPU"


        ELSE (current_player is "CPU"):
            SET valid_move to False
            WHILE valid_move is False:
                RANDOMLY SELECT a row_index
                IF the selected row has no sticks:
                    CONTINUE to the next iteration of the loop
                RANDOMLY SELECT sticks_to_take between 1 and the number of sticks in the selected row
                SET valid_move to True

            ADD a short delay
            REMOVE sticks_to_take from the selected row in sticks
            PRINT move made by the CPU
            SET current_player to "User"

    PRINT "Game Over!"
    IF current_player is "User":
        INCREMENT cpu_wins by 1
        PRINT "CPU wins!"
    ELSE:
        INCREMENT user_wins by 1
        PRINT user_name + " wins!"

    PROMPT user to play again
    IF user wants to play again:
        SET play_again to True
    ELSE:
        PRINT scores for user_name and CPU
        PRINT "Thanks for playing!"
        SET play_again to False.

PROGRAM END

FUNCTION print_move(player, row_index, sticks_taken):
    PRINT player + " took " + sticks_taken + " stick(s) from row " + (row_index + 1)
```