```
BEGIN

INITIALIZE game window with screen_width = 820 and screen_height = 600
LOAD board image and Snakes and Ladders logo

INITIALIZE game variables:
    screen = create display window with screen_width and screen_height
    set window caption to "Snake and Ladder Game"
    colors = define RGB values for different colors
    font = initialize default font with size 36

DEFINE generate_ladders_position():
    Initialize empty list for ladder positions
    FOR i = 1 to 14 DO:
        REPEAT:
            Generate random position for ladder start
        UNTIL position does not conflict with other ladders
        Add position to ladder list
    RETURN ladder positions

DEFINE generate_snakes_position(ladders_list):
    Initialize empty list for snake positions
    FOR i = 1 to 10 DO:
        REPEAT:
            Generate random position for snake start
        UNTIL position does not conflict with ladders and other snakes
        Add position to snake list
    RETURN snake positions

DEFINE get_players_list():
    Initialize empty list for player names
    FOR i = 1 to 2 DO:
        Initialize input box for player name
        WHILE player name not entered:
            FOR each event in event queue:
                IF event type is QUIT:
                    EXIT game
                IF event type is MOUSEBUTTONDOWN:
                    Check if input box is clicked
                    TOGGLE input box active state
                IF event type is KEYDOWN AND input box is active:
                    IF key is ENTER:
                        END input
                    ELSE IF key is BACKSPACE:
                        Remove last character from input
                    ELSE:
                        Add character to input
            Clear screen with background color
            Display prompt and input box
            UPDATE display
        Add player name to list
    RETURN player names

DEFINE get_cell_num(cell):
    Calculate row and column based on cell number
    RETURN row and column

DEFINE draw_board(ladders_list, snakes_list, player_name, dice_value, position, player_color):
    Clear screen with background color
    Draw background image and logo
    Draw dice with current value
    Display current player's turn and position
    FOR each snake:
        Calculate start and end cell
        Draw snake as a red line
    FOR each ladder:
        Calculate start and end cell
        Draw ladder as a black line
    Draw roll button
    UPDATE display

DEFINE draw_players(positions, colors):
    FOR each player:
        IF player is at starting position:
            CONTINUE
        Calculate cell position
        Draw player as a circle

DEFINE draw_dice(dice_value):
    Create surface for dice
    Fill surface with white color
    Draw dice border
    Draw dots based on dice value
    Draw dice on screen

DEFINE roll_dice():
    RETURN random dice value between 1 and 6
```

```
DEFINE check_for_ladder(position, ladders_list, player_name):
    IF position in ladders_list:
        Move player up the ladder
        RETURN new position and message
    RETURN position and no message

DEFINE check_for_snake(position, snakes_list, player_name):
    IF position in snakes_list:
        Move player down the snake
        RETURN new position and message
    RETURN position and no message

DEFINE show_message(msg):
    Render message text
    Clear previous message area
    Draw new message
    UPDATE display

DEFINE draw_restart_button():
    Create rectangle for button
    Draw button rectangle
    Render button text
    Center text on button
    Draw text on button
    UPDATE display
    RETURN button rectangle

MAIN:
    player_names = get_players_list()
    colors = assign colors to players
    Clear screen with white color
    ladders_list = generate_ladders_position()
    snakes_list = generate_snakes_position(ladders_list)
    positions = initialize player positions
    current_player = 0
    draw_board(ladders_list, snakes_list, player_names[current_player], 1, 0,
colors[player_names[current_player]])
    draw_players(positions, colors)
    draw_dice(1)
    UPDATE display
    game_over = False

    WHILE True:
        roll = False
        FOR each event in event queue:
            IF event type is QUIT:
                EXIT game
            IF event type is MOUSEBUTTONDOWN:
                mouse = GET mouse position
                IF roll button is clicked:
                    roll = True
                IF game_over AND restart button is clicked:
                    RESTART game

        IF roll AND NOT game_over:
            player_name = player_names[current_player]
            position = positions[player_name]
            dice_value = roll_dice()
            new_position = position + dice_value
            IF new_position > 100:
                new_position = 100
            new_position, ladder_msg = check_for_ladder(new_position, ladders_list, player_name)
            new_position, snake_msg = check_for_snake(new_position, snakes_list, player_name)
            positions[player_name] = new_position
            draw_board(ladders_list, snakes_list, player_name, dice_value, new_position,
colors[player_name])
            draw_players(positions, colors)
            IF ladder_msg:
                show_message(ladder_msg)
            ELSE IF snake_msg:
                show_message(snake_msg)
            ELSE:
                show_message("")
            IF new_position == 100:
                show_message(player_name + " won the game!")
                UPDATE display
                WAIT 1 second
                restart_button = draw_restart_button()
                game_over = True
            current_player = (current_player + 1) % 2
        UPDATE display

IF __name__ == "__main__":
    MAIN()
```