



# Silly Slots

TERMINAL CASINO

Chinmay K Shetty | 2024 | C Language

Contents

Abstract: .....2

Introduction: .....2

System Design: .....2

Implementation: .....2

Testing and Results:.....3

Output Screenshots: .....3

Conclusion:..... 10

References: ..... 11

## Abstract:

This project is a text-based casino game written entirely in C. It offers five unique games: Number Hunt, Number Ambush, Coded Danger, High Card (2 player), and JackPot. Each game features humorous text descriptions and unexpected twists, prioritizing entertainment over potential winnings. The program allows players to manage virtual deposits and withdrawals within the lighthearted casino theme.

## Introduction:

The Silly Slots Terminal Casino is a game program designed to provide a humorous and entertaining experience for users. Unlike traditional casino games, it prioritizes laughter over potential financial gain. It achieves this through witty text descriptions, unexpected gameplay elements, and a focus on fun rather than monetary rewards.

## System Design:

The program is built on a menu-driven system. Users are presented with a list of available games and can choose the one they want to play. Each game has its own set of rules and logic implemented using C functions. The program keeps track of the player's virtual balance, allowing deposits and withdrawals within the game's framework.

## Implementation:

The programming language used for this project is C. C offers a good balance of control, efficiency, and ease of use for developing text-based games.

The program likely utilizes various C functions for:

- **User input handling** (e.g., `scanf` or `getchar`)
- **Random number generation** (e.g., `rand`)
- **String manipulation** (e.g., `strcpy`, `strcat`)
- **Conditional statements** (e.g., `if`, `else`)
- **Loops** (e.g., `for`, `while`)
- **User-defined functions:** The program likely utilizes separate functions for each game, promoting code modularity and reusability. These functions encapsulate the game's logic, making the code easier to understand and maintain.
- **User-defined data types (structures):** Structures can be used to create custom data types that group related variables under a single name. For example, a `Player` structure could store information like name, balance, and current game state.

- **Pointers:** Pointers allow indirect memory access, which can be useful for dynamically allocating memory during gameplay or manipulating strings.

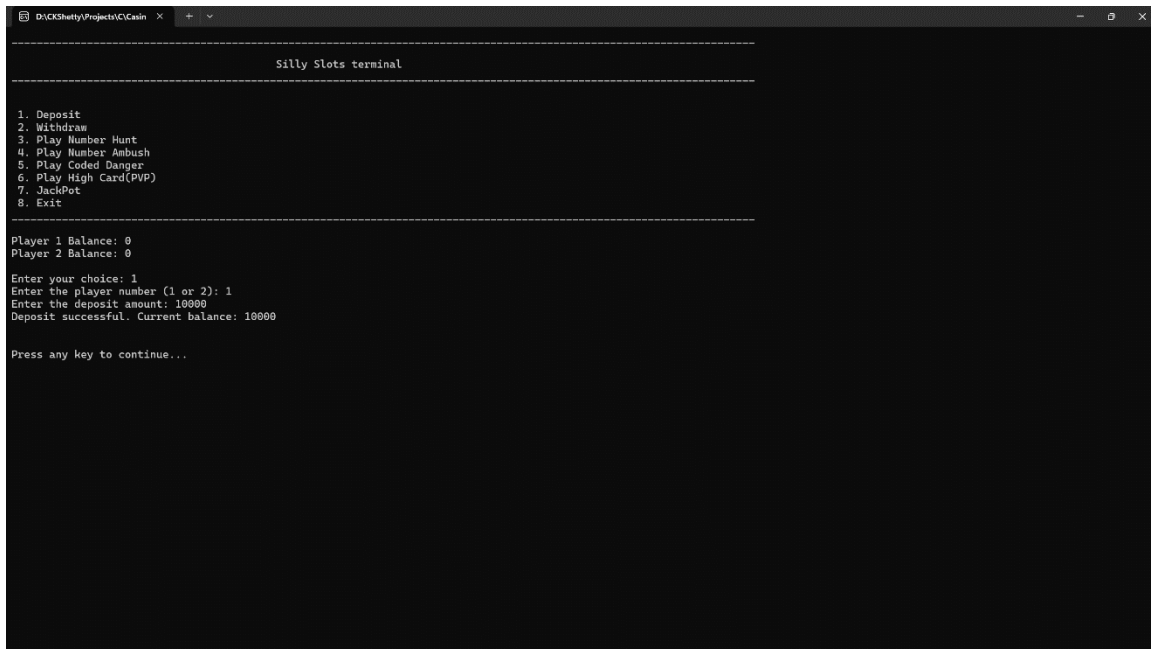
The [Complete Source Code](#) for this project can be found in the my [GitHub Repository](#).

## Testing and Results:

The program is thoroughly tested with various user inputs and across all game options. Sample test cases are documented, including:

- Winning and losing scenarios for each game.
- Testing deposit and withdrawal functionality.
- Verifying random number generation for fair gameplay.
- Testing user-defined functions to ensure they operate as expected.

## Output Screenshots:



```
-----
Silly Slots terminal
-----
1. Deposit
2. Withdraw
3. Play Number Hunt
4. Play Number Ambush
5. Play Coded Danger
6. Play High Card(PVP)
7. JackPot
8. Exit
-----

Player 1 Balance: 0
Player 2 Balance: 0

Enter your choice: 1
Enter the player number (1 or 2): 1
Enter the deposit amount: 10000
Deposit successful. Current balance: 10000

Press any key to continue...
```

```
O:\CKShetty\Projects\C\Casin x + v
-----
Silly Slots terminal
-----
1. Deposit
2. Withdraw
3. Play Number Hunt
4. Play Number Ambush
5. Play Coded Danger
6. Play High Card(PVP)
7. Jackpot
8. Exit
-----
Player 1 Balance: 10000
Player 2 Balance: 0
Enter your choice: |
```

```
O:\CKShetty\Projects\C\Casin x + v
-----
Number Hunt : RULES
-----
You will play Number Hunt. The rules are as follows:
- The player needs to enter the bet amount.
- System generates a random number form 1 to 10
- Your goal is to guess the number
- If your guess is wrong, you loose bet amount credits. And if your guess is right You win 10% of the bet amount
-----
Player 1 Balance: 9550
Player 2 Balance: 0

Enter the bet amount (minimum 10): 50
Guess a number between 1 and 10:
4
Congratulations! You won 5. Current balance: 9555

Press any key to continue...
```

```
O:\CKShetty\Projects\C\Casin x + v
-----
Number Hunt : RULES
-----
You will play Number Hunt. The rules are as follows:
- The player needs to enter the bet amount.
- System generates a random number form 1 to 10
- Your goal is to guess the number.
- If your guess is wrong, you loose bet amount credits. And if your guess is right You win 10% of the bet amount
-----

Player 1 Balance: 9900
Player 2 Balance: 0

Enter the bet amount (minimum 10): 50
Guess a number between 1 and 10.
4
Sorry, you lost. The number was 9. Current balance: 9850

Press any key to continue...

Do you want to play again? (y/n):
```

```
O:\CKShetty\Projects\C\Casin x + v
-----
Number Ambush : RULES
-----
You will play Number Ambush. The rules are as follows:
- The player needs to enter the bet amount.
- As soon as the player presses the enter key. System starts to generate number form 1 to 100
- If you succeed in stopping the ambush. The number you have stopped will be the percent of profit you gain.
-----

Player 1 Balance: 9555
Player 2 Balance: 0

Enter the player number (1 or 2): 1
Enter the bet amount (minimum 10): 100
Press any key to stop the count. If you fail to stop, you will lose your bet.
Counting starts...
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30
5
Congratulations! You won 30. Current balance: 9585
Do you want to play again? (y/n): |
```

```
O:\CKShetty\Projects\C\Casin x + v
-----
Number Ambush : RULES
-----
You will play Number Ambush. The rules are as follows:
- The player needs to enter the bet amount.
- As soon as the player presses the enter key. System starts to generate number form 1 to 100
- If you succeed in stopping the ambush. The number you have stopped will be the percent of profit you gain.
- If you fail to stop before the ambush you lose the current bet.
-----

Player 1 Balance: 9597
Player 2 Balance: 0

Enter the bet amount (minimum 10): 97
Press any key to stop the count. If you fail to stop, you will lose your bet.
Counting starts...
1 2 3 4 5 6 7 8 9 10
Sorry, you lost. Current balance: 9500
Do you want to play again? (y/n):
```

```
O:\CKShetty\Projects\C\Casin x + v
-----
Coded Danger : RULES
-----
You will play Coded Danger. The rules are as follows:
- The game has a grid of 8x8 cells. 10 mines are placed randomly.
- You will start with 0 revealed cells. Each unrevealed cell contains the number of adjacent mines.
- Your goal is to reveal all non-mine cells. If you reveal a mine, you loose bet amount credits.
- For Each round the reward will increase by 5%.
- You will have the option to withdraw or continue after each move.
- Your bet for each round of the game you continue is same.
- If you choose to withdraw, you will lose the current bet.
-----

Player 1 Balance: 9505
Player 2 Balance: 0

Enter the player number (1 or 2): 1
Enter the bet amount (minimum 10): 100
```

```
O:\CKShetty\Projects\C\Casin x + v
0 1 2 3 4 5 6 7 8
-----
0 | ###100000
1 | ###100000
2 | ###100000
3 | ###100000
4 | ###210000
5 | ###100000
6 | ###200000
7 | ###100000
8 | ###100000
Enter row and column to reveal: 5 2
You won 50 credits.Do you want to withdraw or continue? (0 to withdraw, 1 to continue): 1
```

```
O:\CKShetty\Projects\C\Casin x + v
0 1 2 3 4 5 6 7 8
-----
0 | ###100000
1 | ###100000
2 | ###100000
3 | ###100000
4 | ###210000
5 | ###210000
6 | ###200000
7 | ###100000
8 | ###100000
Enter row and column to reveal: 4 2
Game Over! You hit a mine.
0 1 2 3 4 5 6 7 8
-----
0 | 1 2 1 1 0 0 0 0
1 | * 3 * 1 0 0 0 0
2 | * 5 2 1 0 0 0 0
3 | * * 2 1 0 0 0 0
4 | 3 4 * 2 1 0 0 0
5 | * 2 * 1 0 0 0 0
6 | 2 3 2 2 0 0 0 0
7 | 1 * 2 * 1 0 0 0
8 | 1 1 2 1 1 0 0 0
You lost 100 credits. Your current balance is 9405 credits.
Do you want to play again? (y/n):
```



```
O:\CKShetty\Projects\C\Casin x + v
-----
High Card(PVP) : RULES
-----
You will play High Card(PVP). The rules are as follows:
- Each Player need to place a bet of desired amount and later each of them select a number between 1 to 100.
- The winner is the one who has the number closet to the target number generated randomly by system.
- Winner gets 90% of the other players bet amount.
- If a tie occurs both player get rewarded by 80% of their own bet.
-----

Player 1 Balance: 550
Player 2 Balance: 0

Insufficient balance in player 2 account. Current balance: 0

Press any key to continue...
|
```

```
O:\CKShetty\Projects\C\Casin x + v
-----
High Card(PVP) : RULES
-----
You will play High Card(PVP). The rules are as follows:
- Each Player need to place a bet of desired amount and later each of them select a number between 1 to 100.
- The winner is the one who has the number closet to the target number generated randomly by system.
- Winner gets 90% of the other players bet amount.
- If a tie occurs both player get rewarded by 80% of their own bet.
-----

Player 1 Balance: 550
Player 2 Balance: 1000

Enter player 1's bet: 50
Enter player 2's bet: 100
Welcome to High Card!

Player 1, guess a number between 1 and 100: 25
Player 2, guess a number between 1 and 100: 50

Target number: 29
Player 1 wins!
Reward: 90
Player 1's new balance: 640
Do you want to play again? (y/n): |
```

```
D:\CKShetty\Projects\C\Casin x + v - 0 x

-----
777 JackPot : RULES
-----

You will play 777 JackPot. The rules are as follows:
- The game has a grid of 2x2 slots. Items will be generated randomly after placing bet.
- You will start by placing the bet of your choice.
- After placing the bet the generation will start.
- If the Player hits 777 jackpot he/she will be rewarded 100% of their bet amount.
- If any other match is hit player will receive anywhere from 10% to 50% reward.
- And if no match is hit player loses his/her bet.
-----

Player 1 Balance: 1000
Player 2 Balance: 0

Enter the player number (1 or 2): 1
Enter the bet amount (minimum 10): 100
```

```
D:\CKShetty\Projects\C\Casin x + v - 0 x

-----
777 JackPot : RULES
-----

You will play 777 JackPot. The rules are as follows:
- The game has a grid of 2x2 slots. Items will be generated randomly after placing bet.
- You will start by placing the bet of your choice.
- After placing the bet the generation will start.
- If the Player hits 777 jackpot he/she will be rewarded 100% of their bet amount.
- If any other match is hit player will receive anywhere from 10% to 50% reward.
- And if no match is hit player loses his/her bet.
-----

Player Balance: 650
Current bet: 50

& * 7
7 ? $
0 & ?
Spinning...
|
```

```
O:\CKShetty\Projects\C\Casin x + v
-----
777 JackPot : RULES
-----
You will play 777 JackPot. The rules are as follows:
- The game has a grid of 2x2 slots. Items will be generated randomly after placing bet.
- You will start by placing the bet of your choice.
- After placing the bet the generation will start.
- If the Player hits 777 jackpot he/she will be rewarded 100% of their bet amount.
- If any other match is hit player will receive anywhere from 10% to 50% reward.
- And if no match is hit player loses his/her bet.
-----

Player Balance: 7725
Current bet: 50

$ # 7
& 7 7
@ # 7

Congratulations! You won 500. Current balance: 8225
Do you want to play again? (y/n):
```

```
O:\CKShetty\Projects\C\Casin x + v
-----
777 JackPot : RULES
-----
You will play 777 JackPot. The rules are as follows:
- The game has a grid of 2x2 slots. Items will be generated randomly after placing bet.
- You will start by placing the bet of your choice.
- After placing the bet the generation will start.
- If the Player hits 777 jackpot he/she will be rewarded 100% of their bet amount.
- If any other match is hit player will receive anywhere from 10% to 50% reward.
- And if no match is hit player loses his/her bet.
-----

Player Balance: 1000
Current bet: 100

* & @
# & *
* # *

Sorry, you lost. Current balance: 900
Do you want to play again? (y/n): |
```

## Conclusion:

The Silly Slots Terminal Casino is a successful example of a text-based game program written in C. It provides a unique and entertaining experience for users by combining elements of chance with witty humor. The program effectively utilizes C's functionalities to create a menu-driven system, manage player balance, and implement various game mechanics.

## References:

C Programming Language [Documentation](#).