Palak Choudhary
0801CS221104

LaTeX

OBJECTIVE OF THE PROJECT

1. **User-Interactive Mini-Games:** The program provides a menu-driven interface for users to choose from a variety of mini-games, including Random Number Guess, Tic-Tac-Toe, Hangman, Rock Paper Scissors, Dice Simulator, and a Number Pattern game. The objective is to engage users in playing these games.

2. **Game Implementation:** Each mini-game is implemented with its own game rules and logic. For example:

   - **Random Number Guess:** The user has to guess a randomly generated number.
   - **Tic-Tac-Toe:** A two-player game with a 3x3 board.
   - **Hangman:** The user must guess a word by suggesting letters.
   - **Rock Paper Scissors:** The user plays against the computer for a defined number of rounds.
   - **Dice Simulator:** Simulates rolling a dice with user-specified face numbers.
   - **Number Pattern:** The user needs to identify patterns and predict the next number in a sequence.

3. **Debugging with pdb:** The code utilizes Python's built-in `pdb` (Python Debugger) module for debugging. It sets breakpoints using `pdb.set_trace()` at various points in the code to facilitate debugging and inspection of variables and program flow. This can help identify and fix issues or improve code functionality.

4. **Profiling with cProfile:** The code uses Python's `cProfile` module to profile the performance of the code. Profiling measures the time taken by different functions in the code, helping to identify bottlenecks and optimize the program.

5. **Structured Execution:** The program runs in a structured manner, allowing users to choose and play games repeatedly. Users can play multiple rounds and then decide whether to play again or exit.

6. **Loading and Displaying Profiling Statistics:** The code loads profiling data from a file named 'profile_results.pstats' and displays the profiling statistics in a tabular format, sorted by cumulative time. This allows developers to analyze and optimize code performance.

OUTPUT

PYTHON MINI GAMES

Which game would you like to play?
1. Random Number Guess
2. Tic-Tac-Toe
3. Hangman
4. Rock Paper Scissors
5. Dice Simulator
6. Number Pattern

Enter your choice: 1

RANDOM NUMBER GUESS!

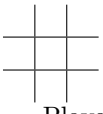
The number to guess is a 2-digit number
Enter your guess: 50
Too low. Guess again!
Enter your guess: 60
Too low. Guess again!
Enter your guess: 80
Too low. Guess again!
Enter your guess: 90
Too high. Guess again!
Enter your guess: 85
Too high. Guess again!
Enter your guess: 83
Right guess!!
Play again? (y/n): y

PYTHON MINI GAMES

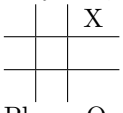
Which game would you like to play?
1. Random Number Guess
2. Tic-Tac-Toe
3. Hangman
4. Rock Paper Scissors
5. Dice Simulator
6. Number Pattern

Enter your choice: 2

TIC-TAC-TOE



Player X, enter a position (0-8): 2



Player O, enter a position (0-8): 3

```
   |   | X
---+---+---
 O |   |
---+---+---
   |   |
```

Player X, enter a position (0-8): 4

```
   |   | X
---+---+---
 O | X |
---+---+---
   |   |
```

Player O, enter a position (0-8): 5

```
   |   | X
---+---+---
 O | X | O
---+---+---
   |   |
```

Player X, enter a position (0-8): 6

```
   |   | X
---+---+---
 O | X | O
---+---+---
 X |   |
```

Player X wins!
Play again? (y/n): y

PYTHON MINI GAMES

Which game would you like to play?
1. Random Number Guess
2. Tic-Tac-Toe
3. Hangman
4. Rock Paper Scissors
5. Dice Simulator
6. Number Pattern

Enter your choice: 3

HANGMAN

```
- - - - -
```

Guess a letter: c
Wrong guess! You have 5 attempts left.

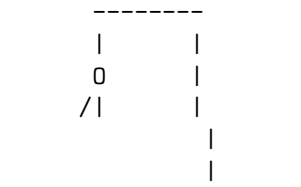
```
  --------
  |       |
  O       |
          |
          |
          |
```

```
- - - - -
```

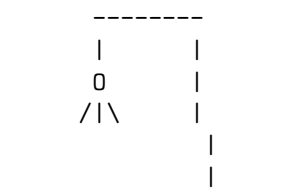
Guess a letter: f
Wrong guess! You have 4 attempts left.

```
  --------
  |       |
  O       |
  |       |
          |
          |
```

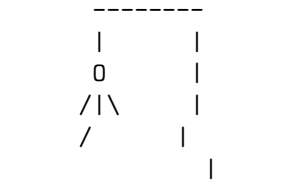
_ _ _ _ _
Guess a letter: a
Wrong guess! You have 3 attempts left.



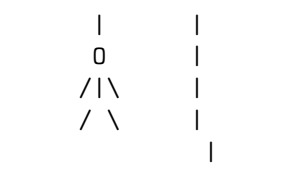
_ _ _ _ _
Guess a letter: d
Wrong guess! You have 2 attempts left.



_ _ _ _ _
Guess a letter: e
_ e _ _ e
Guess a letter: l
Wrong guess! You have 1 attempt left.



_ e _ _ e
Guess a letter: b
Wrong guess! You have 0 attempts left.



Out of attempts. The word was: merge
Play again? (y/n): y

PYTHON MINI GAMES

Which game would you like to play?
1. Random Number Guess
2. Tic-Tac-Toe
3. Hangman
4. Rock Paper Scissors
5. Dice Simulator
6. Number Pattern

Enter your choice: 4

ROCK PAPER SCISSORS

How many points do you want to play for? 2
Enter your choice: rock/paper/scissors: rock
You won! The computer chose scissors
Points: You: 1 Computer: 0
Enter your choice: rock/paper/scissors: rock
You won! The computer chose scissors
Points: You: 2 Computer: 0
GAME OVER!
You won! Congratulations!
Your points: 2
Computer's points: 0
Play again? (y/n): y

PYTHON MINI GAMES

Which game would you like to play?
1. Random Number Guess
2. Tic-Tac-Toe
3. Hangman
4. Rock Paper Scissors
5. Dice Simulator
6. Number Pattern

Enter your choice: 5

DICE SIMULATOR

Enter the number of faces on your dice:4
Your random choice is: 1
Play again? (y/n): y

PYTHON MINI GAMES

Which game would you like to play?

1. Random Number Guess
2. Tic-Tac-Toe
3. Hangman
4. Rock Paper Scissors
5. Dice Simulator
6. Number Pattern

Enter your choice: 6

Welcome to the Number Pattern Game!

Find the next term in this Powers of 2 pattern:
[1, 2, 4, 8, 16]
Your answer: 32
Correct!

Game Over!
Your final score: 1

Play again? (y/n): n

Bye!