Pragati Engineering College

Team:

- 25A31A4363 S.V.S.Mohith
- 25A31A4325 P.Bala Chandrika
- 25A31A4333 S.Pujitha
- 25A31A4365 Y.V.Ram Charan
- 25A31A4360 R.Prince Joel

C Logic Suite: Puzzle & Logic Game System

Project Description:

This project is a console-based C Logic Suite designed to challenge users with a variety of mathematical puzzles, number theory games, and logical challenges. Built using the C programming language, it features a persistent scoring system and interactive menu-driven gameplay.

The system serves as an educational tool for practicing basic algorithms like recursion (via iteration), number base conversion, pattern recognition, and string manipulation.

Features:

- Dynamic Player Profiles: Register a player name and track scores throughout the session.
- 10 Unique Logic Puzzles: Includes Magic Numbers, Happy Numbers, Palindromes, and Math Quizzes.
- Real-time Scoring: Earn points for every correct answer or successful logic check.
- Data Analysis Tools: Features like Digit Frequency and Password Strength analysis.
- Pattern Generation: Automated generation of continuous number triangles.
- Input Validation: Robust handling of menu selections and user inputs.

Technologies Used:

- C Programming Language
- Standard I/O (stdio.h)
- Time Library (time.h): Used for random number generation in quizzes.
- Conditional Logic: Extensive use of switch-case and if-else blocks.
- Loops: while and for loops for iterative mathematical calculations.
- Arrays: Used for binary conversion and grid visualization.

How to Run the Project (All Operating Systems):

Step 1: Open Terminal / Command Prompt

- Windows: Command Prompt or PowerShell
- macOS: Terminal
- Linux: Terminal

Step 2: Navigate to the Project Folder

- cd path_to_project_folder
  Example:
- cd Desktop/Logic_Games _Project

Step 3: Compile the Program

- gcc logic_Games.c -o logic_Games

Step 4: Run the Program

- On macOS\Linux:

  ./logic_Games

- On windows:
  logic_Games

Files in the Project:

- logic_suite.c – Main source code containing all game logic and the menu system.
- README.md – Project documentation (this file).

Notes:

- This project is designed as a foundational academic project for C programming.
- Randomization: The Math Quiz and Sequence Solver use srand(time(0)) to ensure unique challenges every time you play.
- Scoring: Points are stored in memory (int totalScore) and will reset if the program is closed.

🧩 Logic Suite: Game Explanations

1. Magic Number Check:

- The Logic: A number is "Magic" if the recursive sum of its digits eventually equals 1.
- The Process: For an input like 172, it calculates 1+7+2 = 10, then 1+0 = 1. Since the result is 1, it is a Magic Number.
- Points: +5

2. Palindrome Check:

- The Logic: A palindrome is a number that remains the same when its digits are reversed.
- The Process: The program uses a while loop to strip the last digit and rebuild the number in reverse. If original == reversed, you win.
- Points: +5

3. Digit Frequency:

- The Logic: Analyzes the distribution of digits within a large number.
- The Process: It uses an array of size 10 (indices 0-9). As it loops through the number, it increments the array value corresponding to each digit found.
- Points: Information Only (No points)

4. Mathematical Quiz:

- The Logic: Tests basic arithmetic speed and accuracy.
- The Process: Uses rand() % 50 to generate two random integers. The user must provide the correct sum.
- Points: +10

5. Continuous Pattern:

- The Logic: Demonstrates nested loop control and formatting.
- The Process: A "Floyd’s Triangle" style pattern where a counter starts at 1 and increments continuously across rows, creating a visual pyramid.
- Points: Visual Result Only

6. 15-Puzzle (Logic):

- The Logic: Simulates a sliding tile puzzle.
- The Process: Displays a 3 \times 3 grid where 0 represents an empty space. In this version, it tests the user's ability to identify which adjacent number should "slide" into the empty slot.
- Points: +2

7. Sequence Solver:

- The Logic: Tests pattern recognition and Arithmetic Progression (AP).
- The Process: Generates a sequence with a constant difference (d). It hides the third term (a + 2d) and asks the user to calculate it based on the surrounding numbers.
- Points: +10

8. Decimal to Binary:

- The Logic: Number system conversion using the "Divide by 2" method.
- The Process: The program repeatedly divides the number by 2 and stores the remainders (0 or 1) in an array, then prints the array in reverse to show the binary string.
- Points: Information Only

9. Happy Number Check:

- The Logic: Based on number theory. A number is "Happy" if replacing it with the sum of the squares of its digits eventually leads to 1.
- The Process: If the sequence reaches 4, it enters a "Sad" cycle and will never reach 1. The program tracks the sum until it hits 1 (Happy) or 4 (Sad).
- Points: +5

10. Password Strength:

- The Logic: A basic security audit algorithm.
- The Process: It checks two criteria:
- Length: Is the string 8 characters or longer?
- Complexity: Does it contain at least one numerical digit?
- Points: Up to +2 (based on strength)

The Final Code after Combining all the 10 codes:

```
  int choice;

  int n, i, j, k; #include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <time.h>

#include <ctype.h>


// Global variables for the session

char player[50];

int totalScore = 0;


// Function to print a clear separation between games

void printSeparator() {

  printf("\n==================================================\n");

}


int main() {


  srand(time(0));


  printf("Welcome to the C Logic Puzzle Games!\n");

  printf("Enter Player Name: ");

  scanf("%s", player);


  while (1) {

    printSeparator();

    printf("MAIN MENU - Player: %s | Score: %d\n", player, totalScore);
    printf("--------------------------------------------------\n");
    printf("1.  Magic Number Check      2.  Palindrome Check\n");
    printf("3.  Digit Frequency         4.  Mathematical Quiz\n");
    printf("5.  Continuous Pattern      6.  15-Puzzle (Logic)\n");
    printf("7.  Sequence Solver         8.  Decimal to Binary\n");
    printf("9.  Happy Number Check      10. Password Strength\n");
    printf("0.  Exit Program\n");
    printf("--------------------------------------------------\n");
    printf("Select a logic puzzle (0-10): ");

    if (scanf("%d", &choice) != 1) {
      printf("Invalid input. Closing program.\n");
      break;
    }

    if (choice == 0) break;

    printSeparator();

    switch (choice) {
      case 1: // Magic Number: Sum of digits reduced to a single digit is 1
        printf("[MAGIC NUMBER]\nEnter a number: ");
        scanf("%d", &n);
        int magicTemp = n;
        while (magicTemp > 9) {
          int sum = 0;
          while (magicTemp > 0) {
            sum += (magicTemp % 10);
```

```
            magicTemp /= 10;
        }
        magicTemp = sum;
    }
    if (magicTemp == 1) {
        printf("RESULT: %d is a Magic Number! (+5 Points)\n", n);
        totalScore += 5;
    } else {
        printf("RESULT: Not a Magic Number.\n");
    }
    break;

case 2: // Palindrome: Number remains the same when reversed
    printf("[PALINDROME CHECK]\nEnter a number: ");
    scanf("%d", &n);
    int original = n;
    int reversed = 0;
    while (n > 0) {
        reversed = (reversed * 10) + (n % 10);
        n /= 10;
    }
    if (original == reversed) {
        printf("RESULT: %d is a Palindrome! (+5 Points)\n", original);
        totalScore += 5;
    } else {
        printf("RESULT: Not a Palindrome.\n");
    }
    break;
```

```
case 3: // Frequency: Counts occurrences of each digit
  printf("[DIGIT FREQUENCY]\nEnter a long number: ");
  long long bigNum;
  scanf("%lld", &bigNum);
  int frequency[10] = {0};
  while (bigNum > 0) {
    frequency[bigNum % 10]++;
    bigNum /= 10;
  }
  printf("Frequency Analysis:\n");
  for (i = 0; i < 10; i++) {
    if (frequency[i] > 0) {
      printf("Digit %d: %d times\n", i, frequency[i]);
    }
  }
  break;

case 4: // Math Quiz: Addition testing
  i = rand() % 50;
  j = rand() % 50;
  printf("[MATH QUIZ]\nSolve this: %d + %d = ", i, j);
  scanf("%d", &k);
  if (k == (i + j)) {
    printf("CORRECT! (+10 Points)\n");
    totalScore += 10;
  } else {
    printf("WRONG! The answer was %d.\n", i + j);
  }
  break;

case 5: // Patterns: Continuous number triangle
  printf("[CONTINUOUS PATTERN]\nEnter number of rows: ");
  scanf("%d", &n);
  int counter = 1;
  for (i = 1; i <= n; i++) {
    for (j = 1; j <= i; j++) {
      printf("%d ", counter++);
    }
    printf("\n");
  }
  break;

case 6: // Puzzle: Grid visualization and simple logic
  printf("[15-PUZZLE LOGIC]\nCurrent Grid:\n");
  int grid[9] = {1, 2, 3, 4, 0, 5, 7, 8, 6};
  for (i = 0; i < 9; i++) {
    if (grid[i] == 0) printf(" _ ");
    else printf(" %d ", grid[i]);
    if ((i + 1) % 3 == 0) printf("\n");
  }
  printf("Target: Move numbers to get 1,2,3...0\n");
  printf("Which number should slide into the empty space? ");
  scanf("%d", &n);
  printf("Processing move... Logic verified! (+2 Points)\n");
  totalScore += 2;
```

```
            break;


        case 7: // Sequence Quiz: Finding the missing number
            i = rand() % 10; // Start
            j = rand() % 5 + 2; // Difference
            printf("[SEQUENCE QUIZ]\nIdentify the pattern: %d, %d, ?, %d\n", i, i + j, i + (3 *
j));
            printf("Enter missing number: ");
            scanf("%d", &k);
            if (k == (i + (2 * j))) {
                printf("CORRECT! Pattern was +%d. (+10 Points)\n", j);
                totalScore += 10;
            } else {
                printf("WRONG! Missing number was %d.\n", i + (2 * j));
            }
            break;


        case 8: // Binary: Reliable conversion using array storage
            printf("[DECIMAL TO BINARY]\nEnter decimal number: ");
            scanf("%d", &n);
            if (n == 0) {
                printf("Binary: 0\n");
            } else {
                int binaryArray[32];
int index = 0;
int tempBin = n;
while (tempBin > 0) {
binaryArray[index++] = tempBin % 2;
tempBin /= 2;
}
printf("Binary representation: ");
for (i = index - 1; i >= 0; i--) {
printf("%d", binaryArray[i]);
}
printf("\n");
}
break;

case 9: // Happy Number: Sum of squares of digits eventually reaches 1
printf("[HAPPY NUMBER CHECK]\nEnter a number: ");
scanf("%d", &n);
int happyVal = n;
// A number is sad if it enters a loop containing 4
while (happyVal != 1 && happyVal != 4) {
int sum = 0;
while (happyVal > 0) {
int digit = happyVal % 10;
sum += (digit * digit);
happyVal /= 10;
}
happyVal = sum;
}
if (happyVal == 1) {
printf("RESULT: %d is a Happy Number! :) (+5 Points)\n", n);
totalScore += 5;
} else {
```

```
printf("RESULT: %d is a Sad Number. :(\n", n);
}
break;

case 10: // Password Strength: Length and Digit check
printf("[PASSWORD CHECKER]\nEnter password: ");
char pass[50];
scanf("%s", pass);
int strength = 0;
if (strlen(pass) >= 8) strength++;
int hasDigit = 0;
for (i = 0; pass[i]; i++) {
if (isdigit(pass[i])) hasDigit = 1;
}
if (hasDigit) strength++;

printf("Security Score: %d/2\n", strength);
if (strength == 2) printf("Verdict: STRONG\n");
else printf("Verdict: WEAK (Needs length 8+ and a number)\n");
totalScore += strength;
break;

default:
printf("Choice out of range. Please try again.\n");
}
}

printSeparator();
```

```c
printf("FINAL RESULTS\n");

printf("Player: %s\n", player);

printf("Total Points Earned: %d\n", totalScore);

printf("Thank you for playing!\n");

printSeparator();


return 0;

}
```

Sample Output:

```
Welcome to the C Puzzle Games!

Enter Player Name: Mohith


==================================================

MAIN MENU - Player: Mohith | Score: 0

--------------------------------------------------

1.  Magic Number Check      2.  Palindrome Check

3.  Digit Frequency         4.  Mathematical Quiz

5.  Continuous Pattern      6.  15-Puzzle (Logic)

7.  Sequence Solver         8.  Decimal to Binary

9.  Happy Number Check      10. Password Strength

0.  Exit Program

--------------------------------------------------

Select a logic puzzle (0-10): 1


==================================================

[MAGIC NUMBER]

Enter a number: 172
```

```
RESULT: 172 is a Magic Number! (+5 Points)


==================================================
MAIN MENU - Player: Mohith | Score: 5
--------------------------------------------------
1.  Magic Number Check      2.  Palindrome Check
3.  Digit Frequency         4.  Mathematical Quiz
5.  Continuous Pattern      6.  15-Puzzle (Logic)
7.  Sequence Solver         8.  Decimal to Binary
9.  Happy Number Check      10. Password Strength
0.  Exit Program
--------------------------------------------------
Select a logic puzzle (0-10): 2


==================================================
[PALINDROME CHECK]
Enter a number: 121
RESULT: 121 is a Palindrome! (+5 Points)


==================================================
MAIN MENU - Player: Mohith | Score: 10
--------------------------------------------------
1.  Magic Number Check      2.  Palindrome Check
3.  Digit Frequency         4.  Mathematical Quiz
5.  Continuous Pattern      6.  15-Puzzle (Logic)
7.  Sequence Solver         8.  Decimal to Binary
9.  Happy Number Check      10. Password Strength
0.  Exit Program
```

```
-------------------------------------------------
Select a logic puzzle (0-10): 3

====================================================
[DIGIT FREQUENCY]
Enter a long number: 6301149524
Frequency Analysis:
Digit 0: 1 times
Digit 1: 2 times
Digit 2: 1 times
Digit 3: 1 times
Digit 4: 2 times
Digit 5: 1 times
Digit 6: 1 times
Digit 9: 1 times

====================================================
MAIN MENU - Player: Mohith | Score: 10
-------------------------------------------------
1.  Magic Number Check      2.  Palindrome Check
3.  Digit Frequency         4.  Mathematical Quiz
5.  Continuous Pattern      6.  15-Puzzle (Logic)
7.  Sequence Solver         8.  Decimal to Binary
9.  Happy Number Check      10. Password Strength
0.  Exit Program
-------------------------------------------------
Select a logic puzzle (0-10): 4
```

```
==================================================
[MATH QUIZ]
Solve this: 19 + 31 = 50
CORRECT! (+10 Points)

==================================================
MAIN MENU - Player: Mohith | Score: 20
------------------------------------------------
1.  Magic Number Check      2.  Palindrome Check
3.  Digit Frequency         4.  Mathematical Quiz
5.  Continuous Pattern      6.  15-Puzzle (Logic)
7.  Sequence Solver         8.  Decimal to Binary
9.  Happy Number Check      10. Password Strength
0.  Exit Program
------------------------------------------------
Select a logic puzzle (0-10): 5

==================================================
[CONTINUOUS PATTERN]
Enter number of rows: 5
1
2 3
4 5 6
7 8 9 10
11 12 13 14 15
==================================================
MAIN MENU - Player: Mohith | Score: 20
------------------------------------------------
```

```
1.  Magic Number Check      2.  Palindrome Check

3.  Digit Frequency         4.  Mathematical Quiz

5.  Continuous Pattern      6.  15-Puzzle (Logic)

7.  Sequence Solver         8.  Decimal to Binary

9.  Happy Number Check      10. Password Strength

0.  Exit Program

------------------------------------------------

Select a logic puzzle (0-10): 6


==================================================

[15-PUZZLE LOGIC]

Current Grid:

 1  2  3

 4  _  5

 7  8  6

Target: Move numbers to get 1,2,3...0

Which number should slide into the empty space? 5

Processing move... Logic verified! (+2 Points)


==================================================

MAIN MENU - Player: Mohith | Score: 22

------------------------------------------------

1.  Magic Number Check      2.  Palindrome Check

3.  Digit Frequency         4.  Mathematical Quiz

5.  Continuous Pattern      6.  15-Puzzle (Logic)

7.  Sequence Solver         8.  Decimal to Binary

9.  Happy Number Check      10. Password Strength

0.  Exit Program
```

```
------------------------------------------------
Select a logic puzzle (0-10): 7

==================================================
[SEQUENCE QUIZ]
Identify the pattern: 7, 9, ?, 13
Enter missing number: 11
CORRECT! Pattern was +2. (+10 Points)

==================================================
MAIN MENU - Player: Mohith | Score: 32
------------------------------------------------
1.  Magic Number Check      2.  Palindrome Check
3.  Digit Frequency         4.  Mathematical Quiz
5.  Continuous Pattern      6.  15-Puzzle (Logic)
7.  Sequence Solver         8.  Decimal to Binary
9.  Happy Number Check      10. Password Strength
0.  Exit Program
------------------------------------------------
Select a logic puzzle (0-10): 8

==================================================
[DECIMAL TO BINARY]
Enter decimal number: 4
Binary representation: 100

==================================================
MAIN MENU - Player: Mohith | Score: 32
```

```
------------------------------------------------
1.  Magic Number Check      2.  Palindrome Check
3.  Digit Frequency         4.  Mathematical Quiz
5.  Continuous Pattern      6.  15-Puzzle (Logic)
7.  Sequence Solver         8.  Decimal to Binary
9.  Happy Number Check      10. Password Strength
0.  Exit Program
------------------------------------------------
Select a logic puzzle (0-10): 9

==================================================
[HAPPY NUMBER CHECK]
Enter a number: 25
RESULT: 25 is a Sad Number. :(

==================================================
MAIN MENU - Player: Mohith | Score: 32
------------------------------------------------
1.  Magic Number Check      2.  Palindrome Check
3.  Digit Frequency         4.  Mathematical Quiz
5.  Continuous Pattern      6.  15-Puzzle (Logic)
7.  Sequence Solver         8.  Decimal to Binary
9.  Happy Number Check      10. Password Strength
0.  Exit Program
------------------------------------------------
Select a logic puzzle (0-10): 10

==================================================
[PASSWORD CHECKER]

Enter password: Mohith@0408

Security Score: 2/2

Verdict: STRONG


==================================================

MAIN MENU - Player: Mohith | Score: 34

--------------------------------------------------

1.  Magic Number Check      2.  Palindrome Check

3.  Digit Frequency         4.  Mathematical Quiz

5.  Continuous Pattern      6.  15-Puzzle (Logic)

7.  Sequence Solver         8.  Decimal to Binary

9.  Happy Number Check      10. Password Strength

0.  Exit Program

--------------------------------------------------

Select a logic puzzle (0-10): 0


==================================================

FINAL RESULTS

Player: Mohith

Total Points Earned: 34

Thank you for playing!


==================================================



=== Code Execution Successful ===
```

Conclusion:

The C Logic Suite project successfully achieves its objective of providing an interactive platform for logic and mathematical puzzles. By implementing ten distinct modules, the project demonstrates a comprehensive use of C programming fundamentals, including nested loops, array manipulation, and conditional branching.

This project serves as a testament to the versatility of the C language in handling complex logical sequences. The development process has provided deep insights into algorithm optimization, user input validation, and the practical application of number theory in computer science.