B.M.S COLLEGE OF ENGINEERING BENGALURU
Autonomous Institute, Affiliated to VTU

SPC AAT Report on

VIRTUAL TERMINAL CRICKET

Submitted in partial fulfillment of the requirements for AAT

Bachelor of Engineering
in
Artificial Intelligence and Machine Learning

Submitted by:

CHITRARTH SHARMA
BHASKAR AARYAN

Department of Artificial Intelligence and Machine Learning
B.M.S College of Engineering
Bull Temple Road, Basavanagudi, Bangalore 560 019
2025-2026

B.M.S COLLEGE OF ENGINEERING

DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

DECLARATION

We, Chitrarth Sharma and Bhaskar Aaryan students of 1st Semester, B.E, Department of Artificial Intelligence And Machine Learning, BMS College of Engineering, Bangalore, hereby declare that, this AAT Project entitled "VIRTUAL TERMINAL CRICKET" has been carried out in Department of AI-ML, BMS College of Engineering, Bangalore during the academic semester Oct 2025 – Jan 2026. We also declare that to the best of our knowledge and belief, the AAT Project report is not from part of any other report by any other students.

Student Name **Student Signature**

1. **BHASKAR AARYAN**
2. **CHITRARTH SHARMA**

BMS COLLEGE OF ENGINEERING

DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

CERTIFICATE

This is to certify that the AAT Project titled "**VIRTUAL TERMINAL CRICKET"** has been carried out by Chitrarth Sharma(1BM25AI323-T) and Bhaskar Aaryan(1BM25AI279-T) during the academic year 2025-2026.

Signature of the Faculty in Charge

Table of Contents

1. INTRODUCTION

The "Virtual Terminal Cricket" project is a digital adaptation of the popular childhood game known as "Hand Cricket". Hand cricket is a game where players show scores using their fingers, and if the numbers shown by the batsman and bowler match, the batsman is declared "out." This implementation translates these physical mechanics into a terminal-based simulation using the C programming language.

Project Objectives:

- To simulate a realistic cricket match experience within a command-line interface.
- To implement game logic including a toss system, batting, and bowling phases.
- To utilize random number generation to provide a challenging "System" opponent.

Key Features:

1. **Toss System:** A randomized coin toss where the user chooses "Heads" or "Tails" to decide who bats or bowls first.
2. **Dynamic Scorekeeping:** Real-time updates of runs scored and the total score during each inning.
3. **Input Validation:** Robust error handling to ensure users only enter valid numbers between 0 and 6.
4. **Target Chasing:** A target is set after the first innings, and the second team must chase it to win.

The program utilizes standard C libraries such as `<stdio.h>` for input/output, `<stdlib.h>` for utility functions, and `<time.h>` to seed the random number generator, ensuring that every game played is unique.

2. ALGORITHM

Step 1: Start.

Step 2: Initialize the random number generator using the system clock (`srand(time(NULL))`) to ensure unpredictable gameplay.

Step 3: Perform the Toss Phase:

- **a.** Prompt the user to select Heads (1) or Tails (0).
- **b.** Generate a random value (0 or 1) and compare it with the user's choice.
- **c.** The winner of the toss chooses to either **Bat** or **Bowl**.

Step 4: Execute First Innings:

- **a.** Initialize the score `runs = 0`.
- **b.** Read/Generate the batsman's shot (0-6) and the bowler's delivery (0-6).
- **c. Check for Wicket:** If the shot matches the delivery, the batsman is declared **OUT**.
- **d.** Otherwise, add the shot value to the total `runs` and display the current score.
- **e.** Repeat until the batsman is OUT.

Step 5: Calculate the **Target Score** = (Total runs in 1st Innings) + 1.

Step 6: Execute Second Innings:

- **a.** Initialize `runs = 0` for the second player.
- **b.** Repeat the batting and bowling logic from Step 4.
- **c.** The innings ends if the player is **OUT** or if their `runs` reach or exceed the **Target Score**.

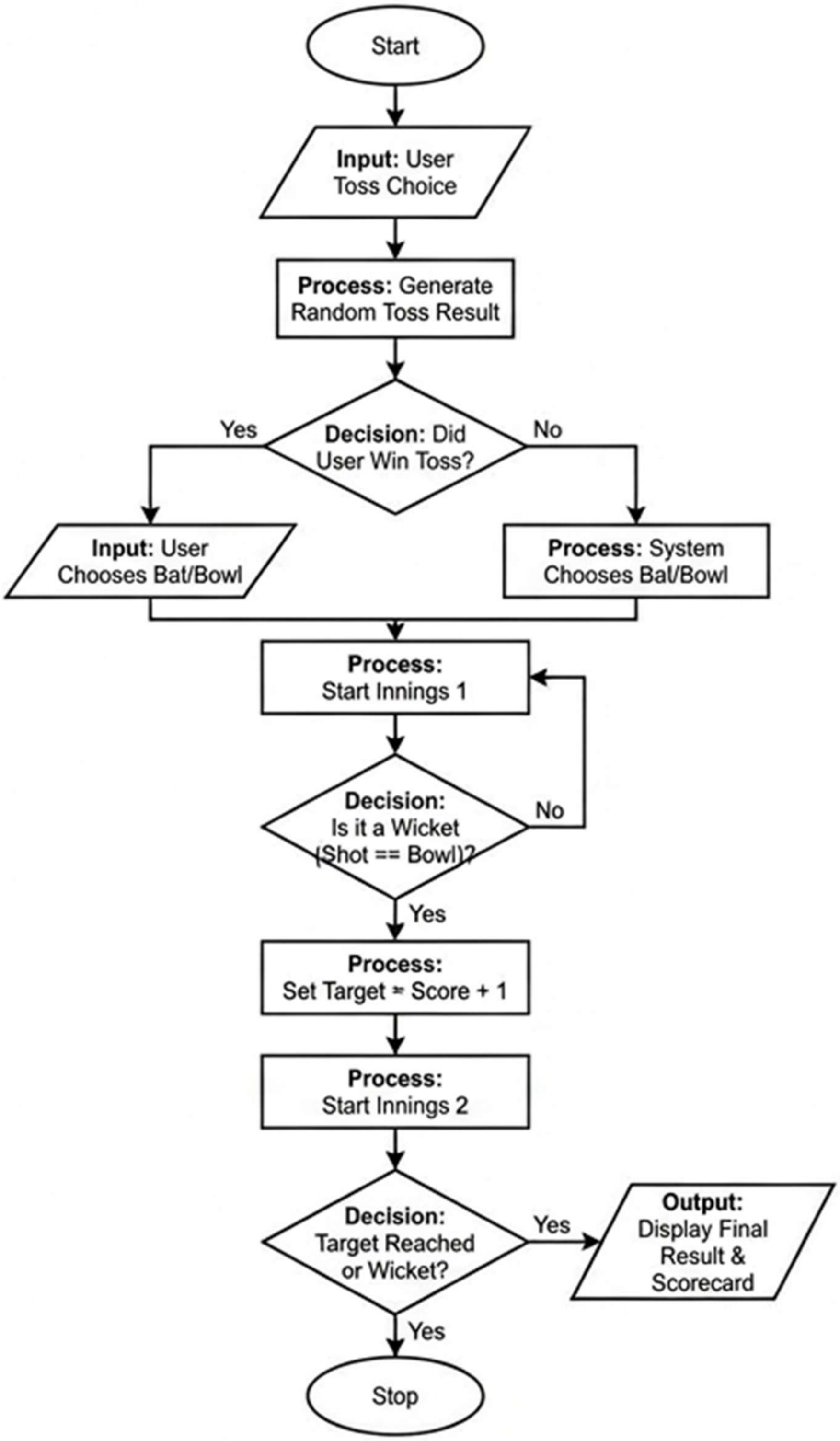
Step 7: Determine the Winner:

- **a.** If the second player reaches the target, display "Second Player Wins."
- **b.** Otherwise, display "First Player Wins."

Step 8: Display the **Final Scorecard** including total runs scored by both teams.

Step 9: Stop

3. FLOWCHART



Start
Input: User Toss Choice
Process: Generate Random Toss Result
Decision: Did User Win Toss?
Yes
No
Input: User Chooses Bat/Bowl
Process: System Chooses Bal/Bowl
Process: Start Innings 1
Decision: Is it a Wicket (Shot == Bowl)?
No
Yes
Process: Set Target = Score + 1
Process: Start Innings 2
Decision: Target Reached or Wicket?
Yes
Output: Display Final Result & Scorecard
Yes
Stop

4. SOURCE CODE

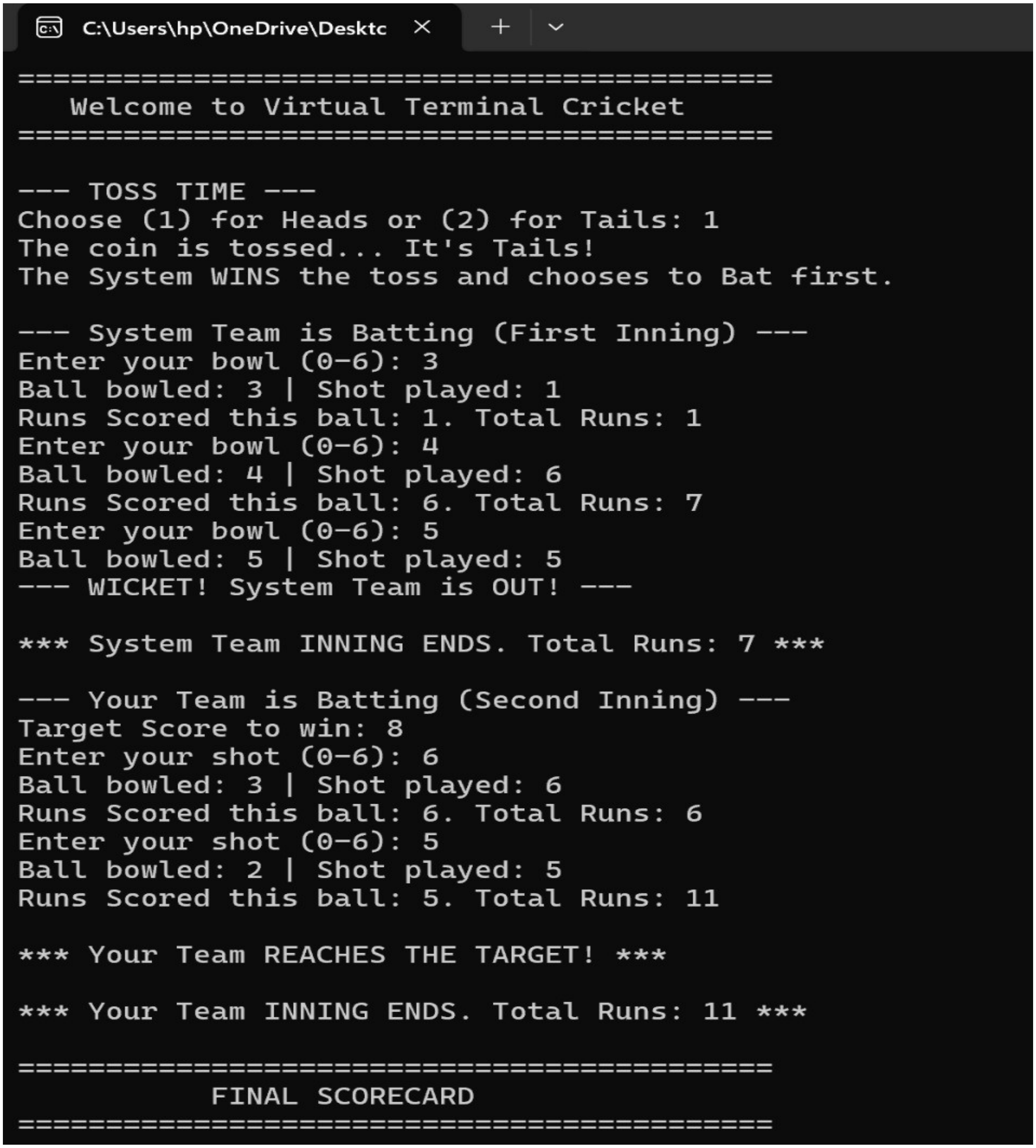
```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
// Function to simulate the first player's inning (User or System)
int first_inning(const char* batsman_name) {
    int runs = 0;
    int ball_batsman;
    int ball_bowler;
    int is_out = 0; // 0 means not out, 1 means out

    printf("\n--- %s is Batting (First Inning) ---\n", batsman_name);
    while (is_out == 0)
        {
        // Decide who is the User for input
        if (batsman_name[0] == 'Y') { // If "Your Team" is batting
            printf("Enter your shot (0-6): ");
            // Input validation loop (a bit complex, but necessary for robustness)
            while (scanf("%d", &ball_batsman) != 1 || ball_batsman < 0 || ball_batsman > 6) {
                printf("Invalid input. Please enter a number between 0 and 6: ");
                while (getchar() != '\n'); // Clear the input buffer
            }
            ball_bowler = rand() % 7; // System bowls a random number
        } else { // If "System Team" is batting
            ball_batsman = rand() % 7; // System bats a random number
            printf("Enter your bowl (0-6): ");
            // Input validation
            while (scanf("%d", &ball_bowler) != 1 || ball_bowler < 0 || ball_bowler > 6) {
                printf("Invalid input. Please enter a number between 0 and 6: ");
                while (getchar() != '\n');
            }
        }

        printf("Ball bowled: %d | Shot played: %d\n", ball_bowler, ball_batsman);
        // Check for OUT condition
        if (ball_batsman == ball_bowler)
            {
            is_out = 1;
            printf("--- WICKET! %s is OUT! ---\n", batsman_name);
        }
        else
            {
            // Add runs (only if the batsman is the one who inputs the runs, which is always ball_batsman)
            runs += ball_batsman;
            printf("Runs Scored this ball: %d. Total Runs: %d\n", ball_batsman, runs);
        }
    }
    printf("\n*** %s INNING ENDS. Total Runs: %d ***\n", batsman_name, runs);
    return runs;
}
// Function to simulate the second player's inning (User or System)
int second_inning(const char* batsman_name, int target_score) {
```

```
int runs = 0;
int ball_batsman;
int ball_bowler;
int is_out = 0; // 0 means not out, 1 means out
printf("\n--- %s is Batting (Second Inning) ---\n", batsman_name);
printf("Target Score to win: %d\n", target_score);

while (is_out == 0 && runs < target_score) {

   // Decide who is the User for input
   if (batsman_name[0] == 'Y') { // If "Your Team" is batting
      printf("Enter your shot (0-6): ");
      // Input validation
      while (scanf("%d", &ball_batsman) != 1 || ball_batsman < 0 || ball_batsman > 6) {
         printf("Invalid input. Please enter a number between 0 and 6: ");
         while (getchar() != '\n');
      }
      ball_bowler = rand() % 7; // System bowls
   } else { // If "System Team" is batting
      ball_batsman = rand() % 7; // System bats

      printf("Enter your bowl (0-6): ");
      // Input validation
      while (scanf("%d", &ball_bowler) != 1 || ball_bowler < 0 || ball_bowler > 6) {
         printf("Invalid input. Please enter a number between 0 and 6: ");
         while (getchar() != '\n');
      }
   }
```

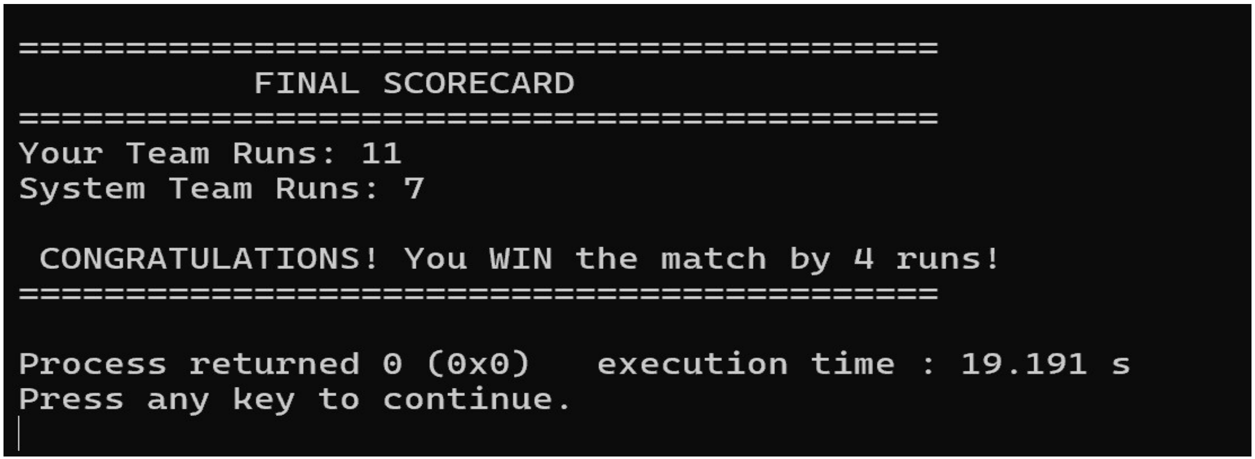
```
        printf("Ball bowled: %d | Shot played: %d\n", ball_bowler, ball_batsman);

        // Check for OUT condition
        if (ball_batsman == ball_bowler) {
            is_out = 1;
            printf("--- WICKET! %s is OUT! ---\n", batsman_name);
        } else {
            runs += ball_batsman;
            printf("Runs Scored this ball: %d. Total Runs: %d\n", ball_batsman, runs);
        }

        // Check if the target is reached after scoring
        if (runs >= target_score && is_out == 0) {
            printf("\n*** %s REACHES THE TARGET! ***\n", batsman_name);
            break;
        }
    }

    printf("\n*** %s INNING ENDS. Total Runs: %d ***\n", batsman_name, runs);
    return runs;
}

// Main function
int main() {
    // Basic setup for randomness

    int user_runs = 0;
int system_runs = 0;
int target = 0;
int toss_choice;
int toss_result;
int user_batting_first = 0; // 0 for false, 1 for true

printf("==========================================\n");
printf("   Welcome to Virtual Terminal Cricket \n");
printf("==========================================\n\n");

// --- 1. TOSS ---
printf("--- TOSS TIME ---\n");
printf("Choose (1) for Heads or (2) for Tails: ");
if (scanf("%d", &toss_choice) != 1 || (toss_choice != 1 && toss_choice != 2)) {
   // Simple error handling
   printf("Invalid input. Defaulting to Heads (1).\n");
   toss_choice = 1;
   while (getchar() != '\n');
}

toss_result = (rand() % 2) + 1; // 1=Heads, 2=Tails

printf("The coin is tossed... It's %s!\n", (toss_result == 1 ? "Heads" : "Tails"));

if (toss_choice == toss_result) {
   printf("You WIN the toss! Choose (1) to Bat or (2) to Bowl: ");
   int choice_after_toss;
```

```
        if (scanf("%d", &choice_after_toss) != 1 || (choice_after_toss != 1 && choice_after_toss != 2)) {
            printf("Invalid input. Defaulting to Bat (1).\n");
            choice_after_toss = 1;
        }
        user_batting_first = (choice_after_toss == 1);
        printf("You chose to %s first.\n", (user_batting_first == 1 ? "Bat" : "Bowl"));
    } else {
        printf("The System WINS the toss and chooses to Bat first.\n");
        user_batting_first = 0; // System always bowls first if they win the toss in this simple version
    }

    // --- 2. GAME INNINGS ---
    if (user_batting_first == 1) {
        // User Bats First
        user_runs = first_inning("Your Team");
        target = user_runs + 1;

        // System Bats Second
        system_runs = second_inning("System Team", target);

    } else {
        // System Bats First
        system_runs = first_inning("System Team");
        target = system_runs + 1;

        // User Bats Second
        user_runs = second_inning("Your Team", target);
    }

    // --- 3. DETERMINE WINNER ---
    printf("\n==============================================\n");
    printf("           FINAL SCORECARD \n");
    printf("==============================================\n");
    printf("Your Team Runs: %d\n", user_runs);
    printf("System Team Runs: %d\n", system_runs);

    if (user_runs > system_runs) {
        printf("\n CONGRATULATIONS! You WIN the match by %d runs! \n", user_runs - system_runs);
    } else if (system_runs > user_runs) {
        printf("\n Tough luck! The System Team WINS the match by %d runs! \n", system_runs - user_runs);
    } else {
        printf("\n The match is a TIE! Well played by both teams! \n");
    }

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

    return 0;
}
```

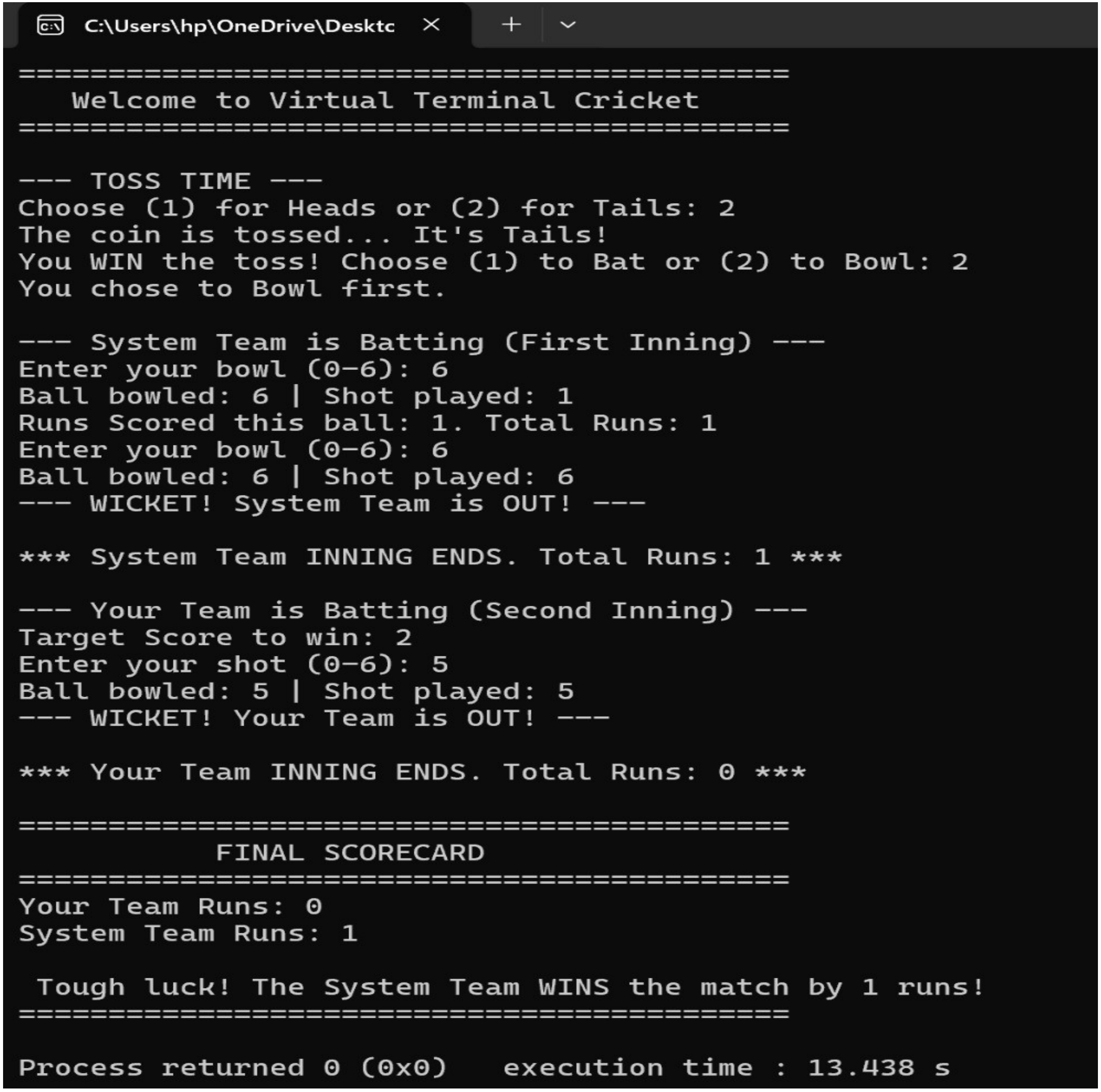
5. RESULTS



```
===========================================
   Welcome to Virtual Terminal Cricket
===========================================

--- TOSS TIME ---
Choose (1) for Heads or (2) for Tails: 1
The coin is tossed... It's Tails!
The System WINS the toss and chooses to Bat first.

--- System Team is Batting (First Inning) ---
Enter your bowl (0-6): 3
Ball bowled: 3 | Shot played: 1
Runs Scored this ball: 1. Total Runs: 1
Enter your bowl (0-6): 4
Ball bowled: 4 | Shot played: 6
Runs Scored this ball: 6. Total Runs: 7
Enter your bowl (0-6): 5
Ball bowled: 5 | Shot played: 5
--- WICKET! System Team is OUT! ---

*** System Team INNING ENDS. Total Runs: 7 ***

--- Your Team is Batting (Second Inning) ---
Target Score to win: 8
Enter your shot (0-6): 6
Ball bowled: 3 | Shot played: 6
Runs Scored this ball: 6. Total Runs: 6
Enter your shot (0-6): 5
Ball bowled: 2 | Shot played: 5
Runs Scored this ball: 5. Total Runs: 11

*** Your Team REACHES THE TARGET! ***

*** Your Team INNING ENDS. Total Runs: 11 ***

===========================================
           FINAL SCORECARD
===========================================
```



```
===========================================
           FINAL SCORECARD
===========================================
Your Team Runs: 11
System Team Runs: 7

 CONGRATULATIONS! You WIN the match by 4 runs!
===========================================

Process returned 0 (0x0)   execution time : 19.191 s
Press any key to continue.
```



```
===========================================
   Welcome to Virtual Terminal Cricket
===========================================

--- TOSS TIME ---
Choose (1) for Heads or (2) for Tails: 2
The coin is tossed... It's Tails!
You WIN the toss! Choose (1) to Bat or (2) to Bowl: 2
You chose to Bowl first.

--- System Team is Batting (First Inning) ---
Enter your bowl (0-6): 6
Ball bowled: 6 | Shot played: 1
Runs Scored this ball: 1. Total Runs: 1
Enter your bowl (0-6): 6
Ball bowled: 6 | Shot played: 6
--- WICKET! System Team is OUT! ---

*** System Team INNING ENDS. Total Runs: 1 ***

--- Your Team is Batting (Second Inning) ---
Target Score to win: 2
Enter your shot (0-6): 5
Ball bowled: 5 | Shot played: 5
--- WICKET! Your Team is OUT! ---

*** Your Team INNING ENDS. Total Runs: 0 ***

===========================================
           FINAL SCORECARD
===========================================
Your Team Runs: 0
System Team Runs: 1

 Tough luck! The System Team WINS the match by 1 runs!
===========================================

Process returned 0 (0x0)   execution time : 13.438 s
```

6. REFERENCES

1. *C Programming Absolute Beginner's Guide* by Greg Perry and Dean Miller.
2. Documentation for `stdlib.h` and `time.h` for random number generation.
3. GeeksforGeeks - C Programming Language.
4. C Programming Made Easy by Dennis Ritchie
5. FreeCodeCamp –(https://www.freecodecamp.org/)
6. https://youtu.be/rQoqCP7LX60?si=eB4q8J05ONpswuVj