CRICKET SCORE DISPLAY

A MINI PROJECT REPORT

Submitted by

CHANDRADEEP KUMAR RAJ

in partial fulfillment for the award of the 1st semester of

Master of Computer Application

In

Department of Computer Science

GREATER NOIDA INSTITUTE OF TECHNOLOGY, GREATER NOIDA

January & 2025

TABLE OF CONTENTS

1. Introduction

Cricket is a sport beloved by millions around the world. It is second most popular game in the world. Most Indians are just crazy about this sport there is multiple application to check cricket scores, it is quite a tough job to maintain a live score of crickets,

But we can create a simple C application to display Cricket score, we can create using basic C knowledge. Keeping track of scores and match statistics is crucial for fans and analysts alike.

The ***Cricket Score Display Program*** is a console-based application developed in C programming language. It is designed to be user-friendly, ensuring that even individuals with minimal technical knowledge can operate it seamlessly.

Through this program, users can select teams, simulate a toss, update and manage scores in real-time, and view detailed match statistics and results.

While numerous applications provide live cricket scores, creating a personalized C application to display cricket scores offers an exciting opportunity to delve into programming using basic C knowledge.

This project aims to develop a simple yet functional *Cricket Score Display application* that simulates live scoring, allowing users to print match statistics, display runs scored, update scores, and show results.

This project addresses this challenge by providing a digital solution that allows users to manage and display cricket scores efficiently.

2. Objectives

The objective of this project is to develop a simple C program that manages and displays cricket match scores, team statistics, and match results through a console-based interface. This program will enable users to select teams, handle the toss, update scores dynamically, and present detailed match statistics and results.

❖ **Develop a Functional Application**:

Build a C program that can display match statistics and updating cricket scores effectively.

❖ **Utilize Basic C Concepts**:

Apply fundamental programming constructs such as variables, data types, control structures, functions, pointer, arrays and structure.

❖ **Interactive User Interface**:

Creating a *console-based Interface* for an intuitive and interactive user interface. Enhance user engagement by incorporating team selection and match result features.

❖ **Enhance Problem-Solving Skills**:

Strengthen logical thinking and code organization through modular programming.

3. Features and Functionalities

Menu-driven options for easy navigation. Utilizes ANSI escape codes for colourful and highlighted text to enhance readability and user-friendly interface.

3.1 Update Score Data

- ❖ **Real-time Updates:** Users can update various aspects of the team's performance in real-time:
  - **Total Runs Scored:** Input the latest runs scored by the team.
  - **Balls Faced:** Update the number of balls faced by the team (with a maximum limit of 120 balls per inning).
  - **Wickets Lost:** Update the number of wickets lost by the team (up to 10 wickets).
  - **Sixes and Fours Hit:** Input the number of sixes and fours hit by the team.
  - **Highest Run Scorer:** Enter the name of the player who scored the highest runs for the team.
  - **Innings Management:** Tracks the status of each inning (completed or ongoing). Automatically switches innings based on user inputs.

The program validates the inputs to ensure data integrity and consistency. For example, it checks that the total score is consistent with the number of sixes and fours.

3.2 Print Runs Scored

- ❖ Shows the current runs scored by each team.

3.3 Print Match Statistics

- ❖ Provides detailed statistics for each team, including runs, wickets, sixes, fours, and highest run scorer.

3.4 Show Match Result

- ❖ Displays the result, indicating the winning team or if the match is ongoing or tied.

3.5 Some Extra Features and Functionalities

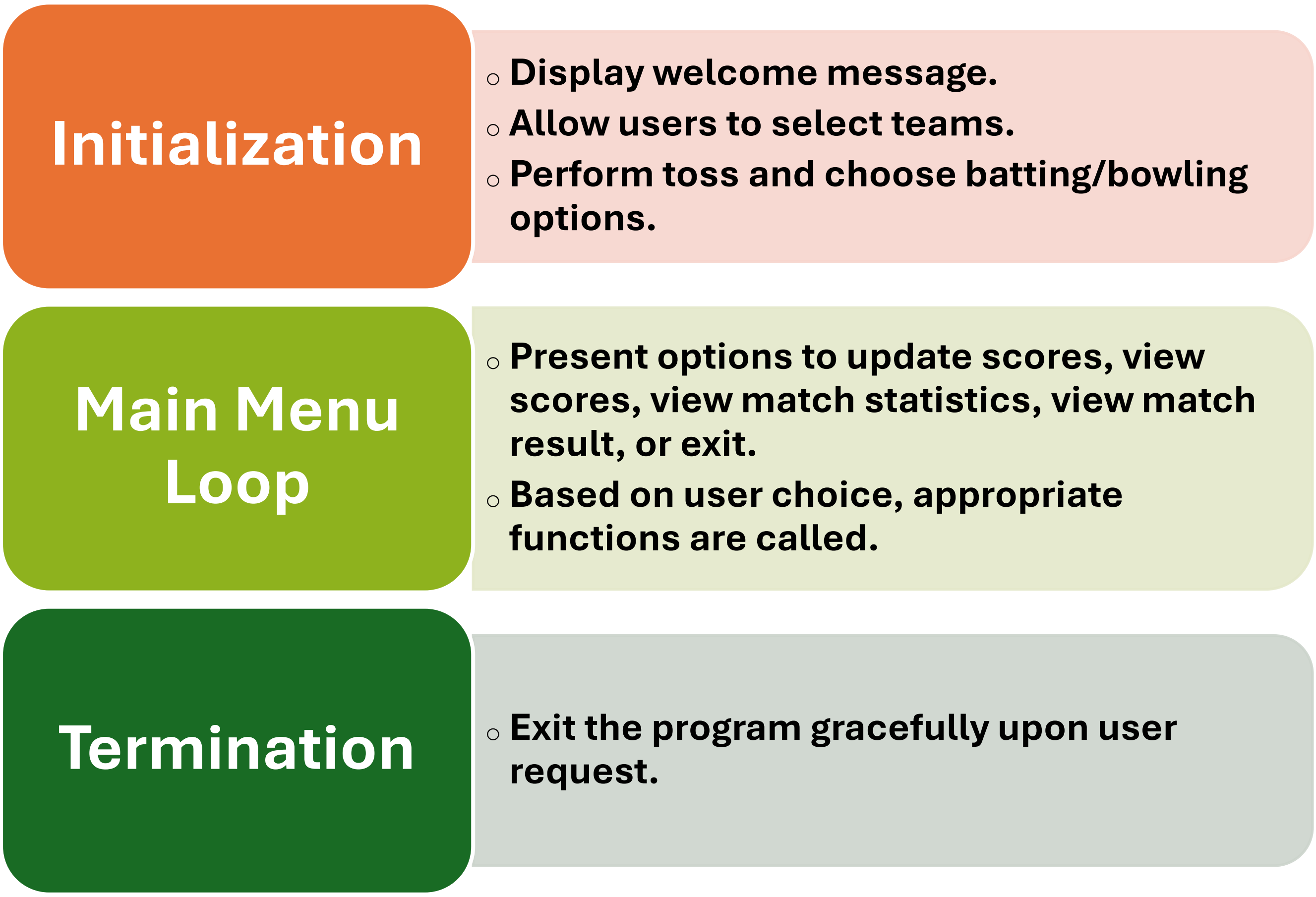
- **Team Selection:** Users can select Team A and Team B from a list of predefined international cricket teams. The program ensures that both teams are distinct to avoid conflicts.
- **Toss Simulation:** Simulates a coin toss to determine which team wins the toss. The toss-winning team can choose to bat or bowl first.

4. System Design and Architecture

The program follows a modular design, dividing functionalities into separate functions for better readability and maintenance. The use of structures helps in organizing data related to teams, ensuring that all relevant information is encapsulated within appropriate data types.

4.1 Architecture Overview

- **High-Level Flow:**



Initialization

- Display welcome message.
- Allow users to select teams.
- Perform toss and choose batting/bowling options.

Main Menu Loop

- Present options to update scores, view scores, view match statistics, view match result, or exit.
- Based on user choice, appropriate functions are called.

Termination

- Exit the program gracefully upon user request.

4.2 Data Structures

❖ **Team Structure (*struct Team*):** The **Team** structure is used to store information about each team's performance, including team name, score, balls faced, wickets, sixes, fours, highest run scorer, and innings status

❖ **Global Variables:**

- **teamlist[10][13]:** Array of predefined team names.
- **batting_team[13], bowling_team[13]:** Stores the names of the batting and bowling teams.
- **teamA, teamB:** Instances of struct Team representing both teams.

4.3 Module Descriptions

❖ **Initialization Module**:

- Sets up match details, including teams , who win the toss.
- Initializes team statistics to default values.

❖ **Score Update Module**:

- Handles inputting of runs, extras, and wickets per ball.
- Updates team statistics accordingly.
- Manages ball counts and 6’s or 4’s hit count.

❖ **Match Statistics Display Module**:

- Provides functions to display current match statistics.
- Shows scores, wicket and ball of each team.

❖ **Result Calculation Module**:

- Determines the match outcome based on the final scores.
- Prepares a match highlight .i.e win by how much runs or tie.

- **User Interface Module**:
    - Manages the menu-driven interface.
    - Handles user inputs and navigates between different functionalities.

5. Source Code Development Plan

5.1 Development Environment

- **Programming Language**: C
- **Compiler**: GCC or any standard C compiler
- **IDE/Text Editor**: Code::Blocks.
- **Development Steps:**
    - **Define Data Structures**: Implement the Team structures as per the system design.
    - **Create the Main Menu**: Develop a user-friendly menu for navigating the application's functionalities.

5.2 Header Files and Constants

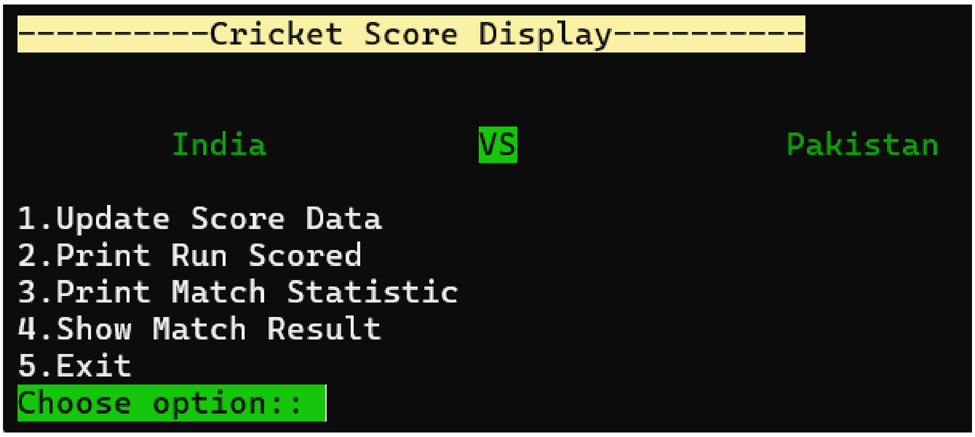
- The program includes essential header files (*stdio.h, string.h, stdlib.h*).
- Constants for text formatting (like colors and highlights) are defined using **#define**.

5.3 Functions

- ***selectteam():*** This function displays the list of available teams and allows the user to select a team by entering the corresponding number.
- ***toss_check():*** This function simulates a coin toss and allows the user to select which team won the toss.
- ***choice_after_toss_win(int toss_value):*** This function lets the toss-winning team choose to bat or bowl.
- ***update_score_data(struct Team *teamX):*** This function updates the score, balls faced, wickets, sixes & fours hit by a team, highest scorer of the team and validates the inputs.
- ***print_run_scored(struct Team teamX):*** This function prints the runs scored by a team.
- ***print_match_statistics(struct Team team):*** This function displays detailed match statistics for a team.
- ***show_match_result(struct Team teamA, struct Team teamB):*** This function displays the final result of the match, indicating the winning team or if the match is ongoing or tied.
- ***main():*** The main function initializes the program, handles team selection, toss, score updates, and calls other functions based on user input.

6. User Interface Design

❖ Main Menu:



```
----------Cricket Score Display----------


        India          VS               Pakistan

1.Update Score Data
2.Print Run Scored
3.Print Match Statistic
4.Show Match Result
5.Exit
Choose option::
```

❖ Submenus:

- For updating scores, the program presents options to update specific data fields.
- Inputs are requested with clear prompts, and instructions are provided for valid input ranges.

❖ Visual Enhancements:

- **Text Formatting:** Bold and coloured text for headings and important information.
- **Highlighting:** Specific highlights for errors, options, and headings to improve readability.

❖ Error Messages:

- Informative error messages guide the user to provide valid inputs.
- Example: "**Invalid Input ! Enter Again !**"

7. Testing and Validation

7.1 Test Cases and Results

- **Team Selection Test:** Selected the same team for both Team A and Team B.

  Result: Program prompted to reselect Team B.

- **Toss and Choice Test:** Chose an invalid option for toss-winning team.

  Result: Received an error message and prompted to choose again.

- **Update Test:** Entered balls faced exceeding 120.

  Result: Program displayed an error and requested valid input.

- **Match Statistics Display Test:** Viewed match statistics without entering the highest run scorer.

  Result: Program highlighted that input was not provided.

- **Match Result Test:** Tested scenarios where the match is ongoing, tied, or one team wins.

  Result: Program correctly displayed the match result based on current data.

❖ **Observations:**

- The program effectively manages user inputs and maintains data integrity.

- Error handling enhances user experience by preventing invalid data entries.

- The visual enhancements make the interface engaging and easy to navigate.

7.2 Validation and Error Handling

❖ **Team Selection Conflict:**

- Ensures that Team A and Team B are not the same.
- Prompts the user to reselect if a conflict is detected.

❖ **Input Validation:**

- Score and Ball Updates:
  - ➢ Validates that the number of balls does not exceed the maximum (120 per inning).
  - ➢ Ensures that the total score is consistent with the number of sixes and fours.

❖ **Wicket Count:** Validates that wickets lost do not exceed 10.

❖ **Innings Status:** Allows only valid options for innings completion status.

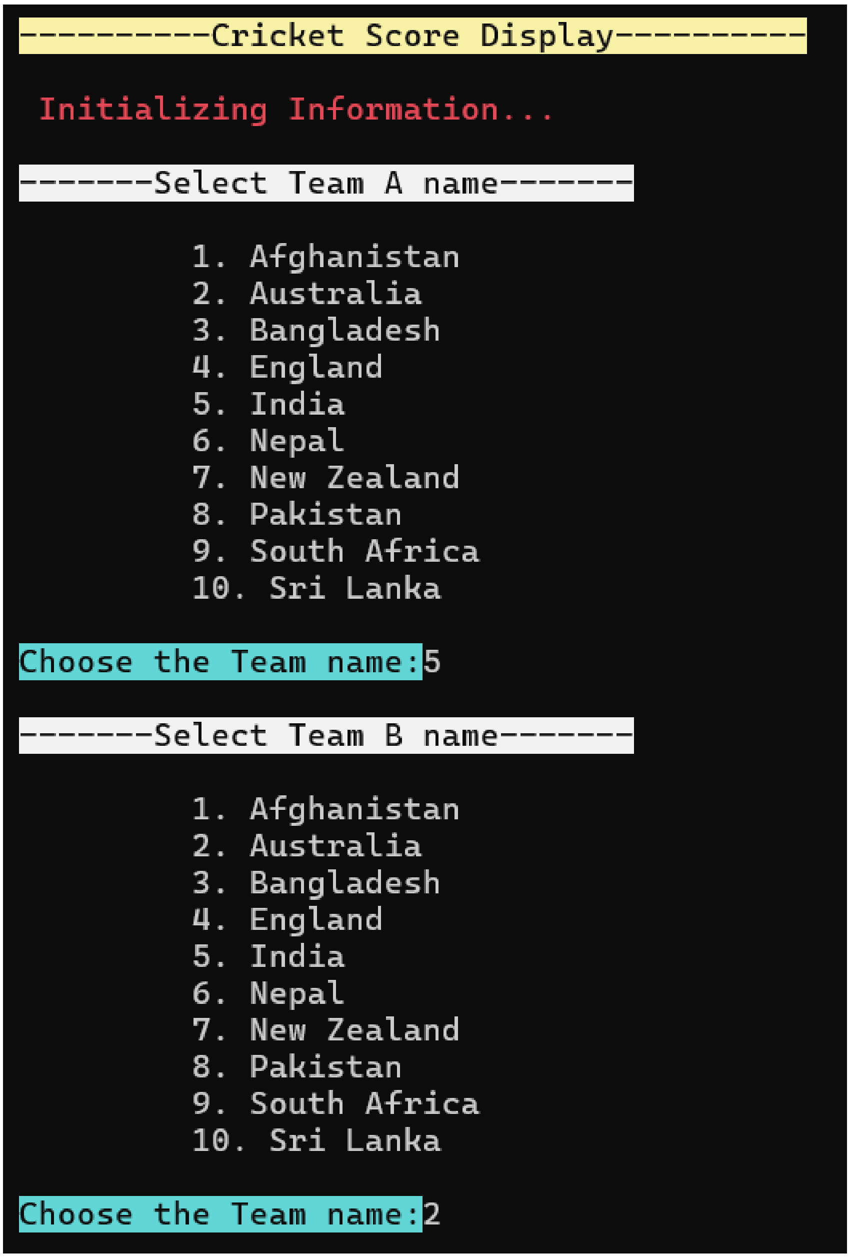
❖ **Error Prompts:**

- Highlights invalid inputs and provides feedback for correction.
- Uses distinctive colouring to draw attention to errors.

8. User Interface Guide

❖ **Initialize the Program:**

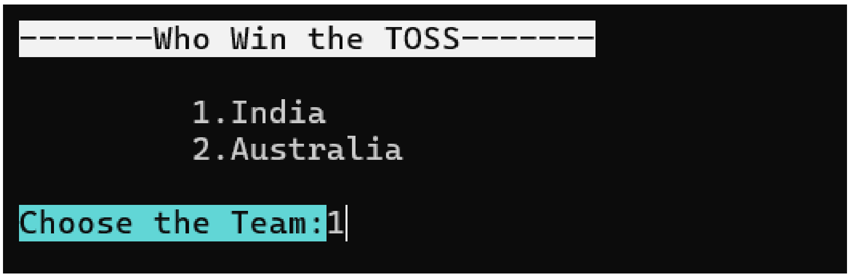
- The program starts with a welcome message and prompts the user to select Team A and Team B from the predefined list.
- It ensures that both selected teams are different.



```
----------Cricket Score Display----------

 Initializing Information...

-------Select Team A name-------

         1. Afghanistan
         2. Australia
         3. Bangladesh
         4. England
         5. India
         6. Nepal
         7. New Zealand
         8. Pakistan
         9. South Africa
         10. Sri Lanka

Choose the Team name:5

-------Select Team B name-------

         1. Afghanistan
         2. Australia
         3. Bangladesh
         4. England
         5. India
         6. Nepal
         7. New Zealand
         8. Pakistan
         9. South Africa
         10. Sri Lanka

Choose the Team name:2
```

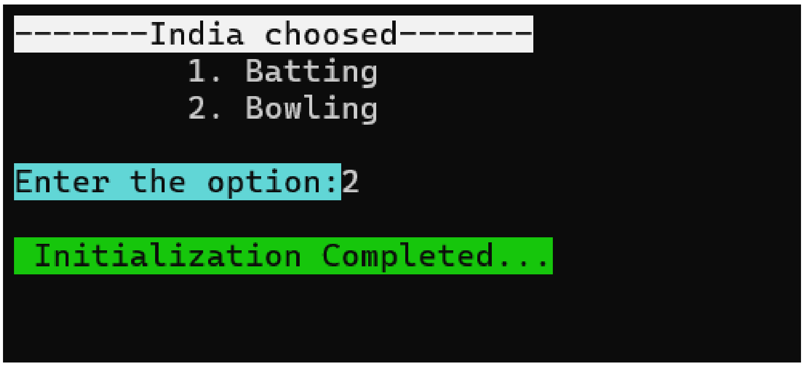
❖ **Toss Handling:**

- The program simulates a coin toss to decide the toss-winning team.



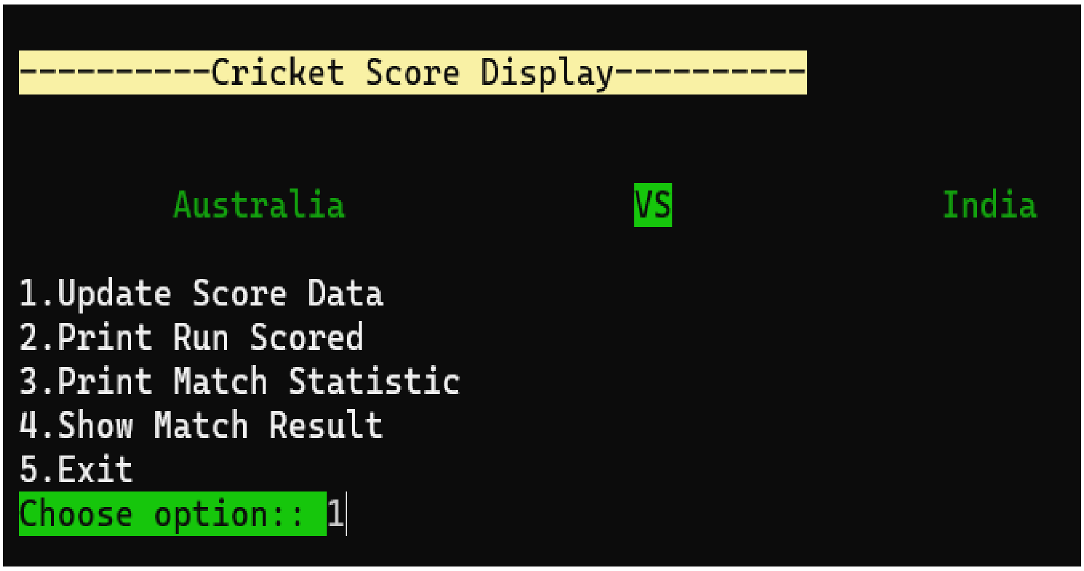
```
-------Who Win the TOSS-------

         1.India
         2.Australia

Choose the Team:1
```

- The toss-winning team chooses to either bat or bowl first.



```
-------India choosed-------
         1. Batting
         2. Bowling

Enter the option:2

 Initialization Completed...
```

❖ Main-Menu:

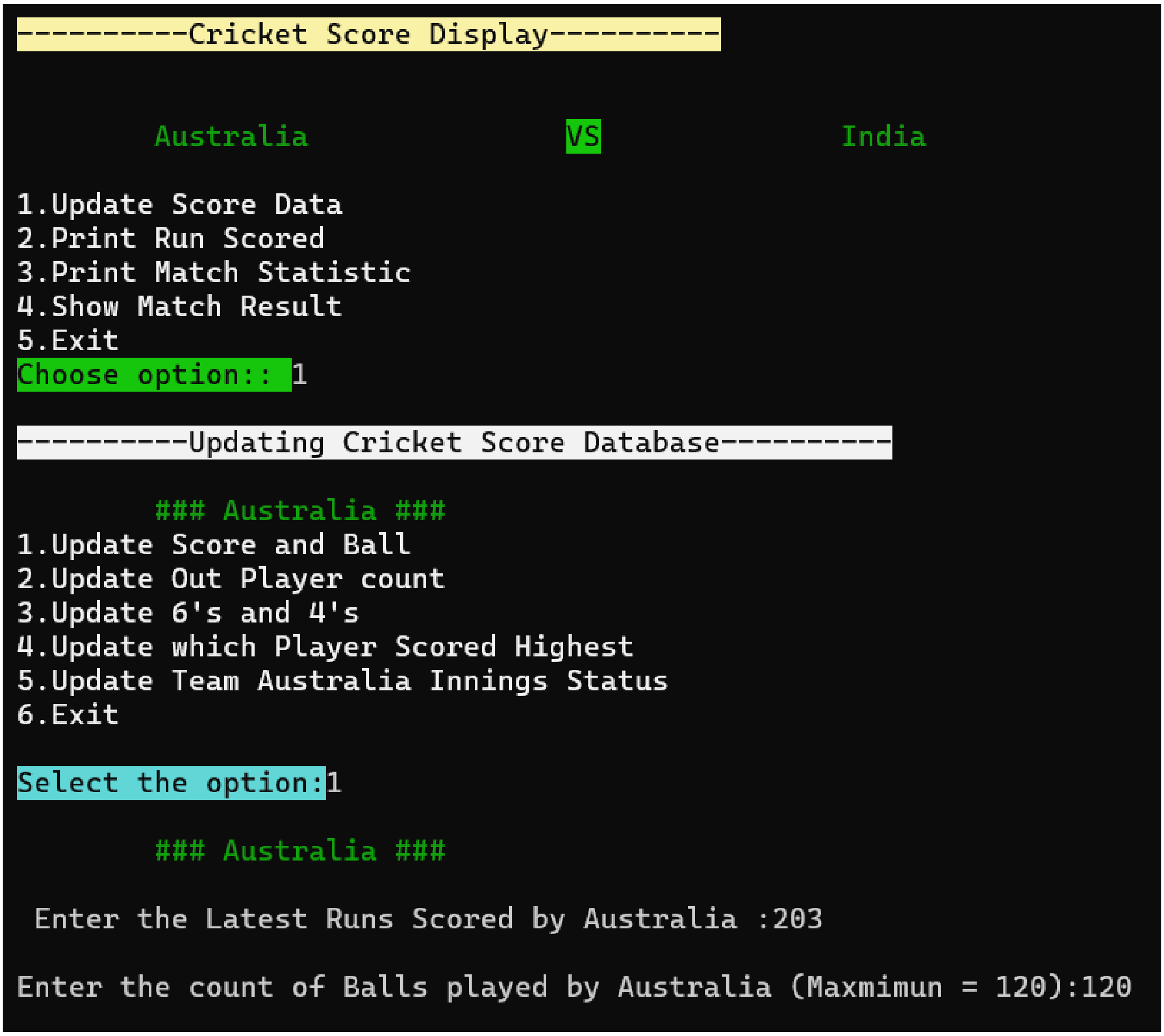


```
----------Cricket Score Display----------


       Australia                 VS              India

1.Update Score Data
2.Print Run Scored
3.Print Match Statistic
4.Show Match Result
5.Exit
Choose option:: 1
```

❖ Score Updates:

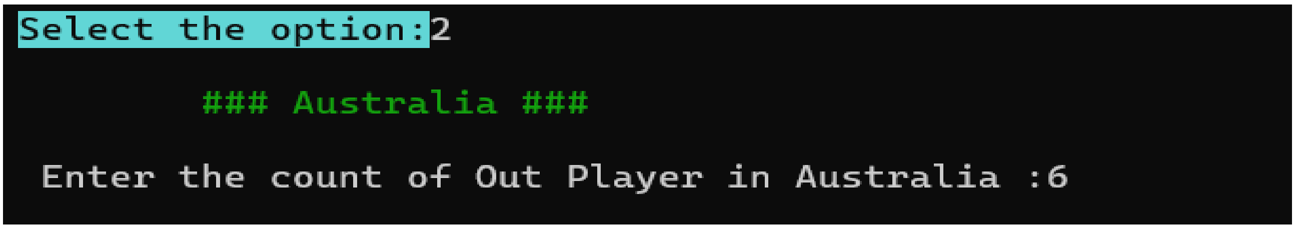
- Users can update the score, balls faced, wickets lost, sixes, and fours hit by the batting team during each inning.
- The program validates the inputs and ensures data consistency.

- **Option 1:** Updating Cricket Score Database



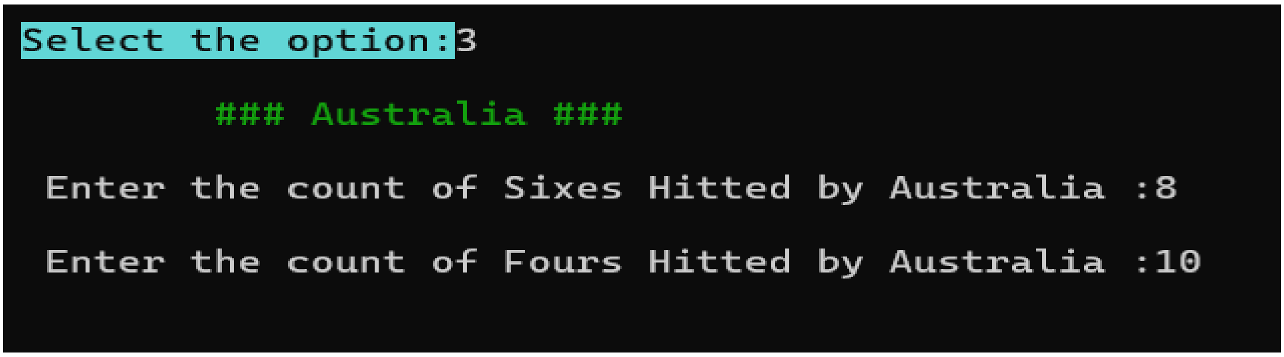
```
----------Cricket Score Display----------


        Australia                    VS                India

1.Update Score Data
2.Print Run Scored
3.Print Match Statistic
4.Show Match Result
5.Exit
Choose option:: 1

----------Updating Cricket Score Database----------

        ### Australia ###
1.Update Score and Ball
2.Update Out Player count
3.Update 6's and 4's
4.Update which Player Scored Highest
5.Update Team Australia Innings Status
6.Exit

Select the option:1

        ### Australia ###

 Enter the Latest Runs Scored by Australia :203

Enter the count of Balls played by Australia (Maxmimun = 120):120
```

- **Option 2:** Enter the data of lost wicket



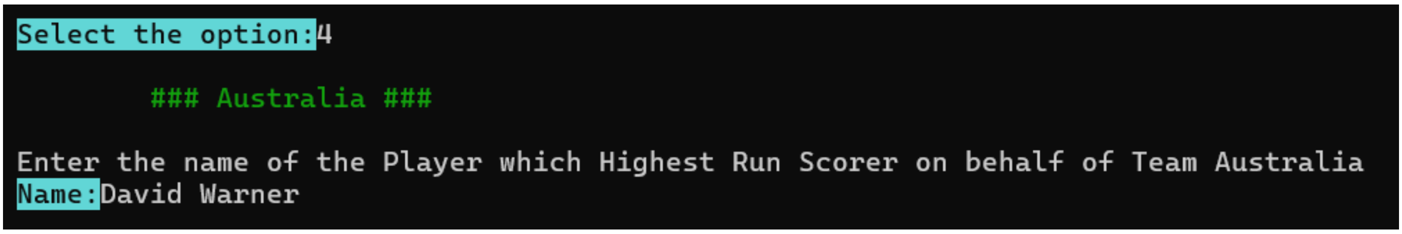
```
Select the option:2

        ### Australia ###

 Enter the count of Out Player in Australia :6
```

- **Option 3:** Enter the 6’s and 4’s Hit count



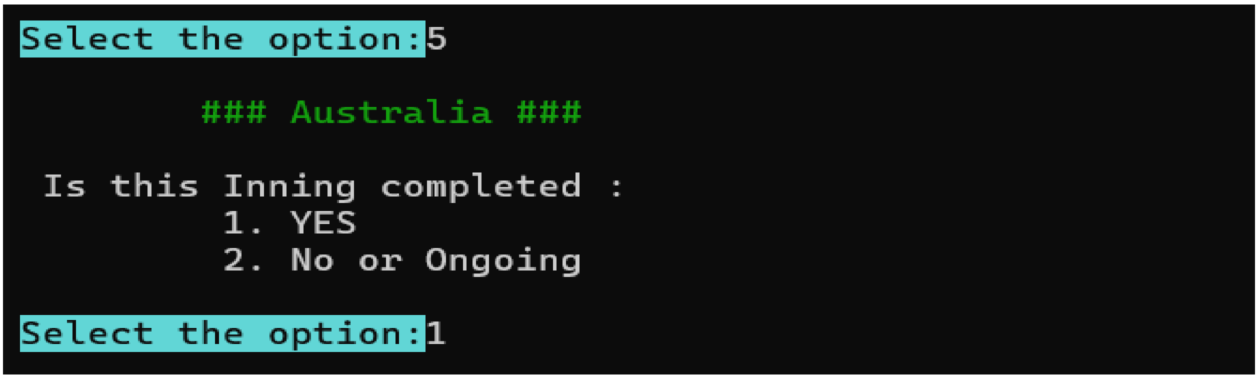
```
Select the option:3

        ### Australia ###

 Enter the count of Sixes Hitted by Australia :8

 Enter the count of Fours Hitted by Australia :10
```

- **Option 4:** Selecting the Highest Run Scorer Player



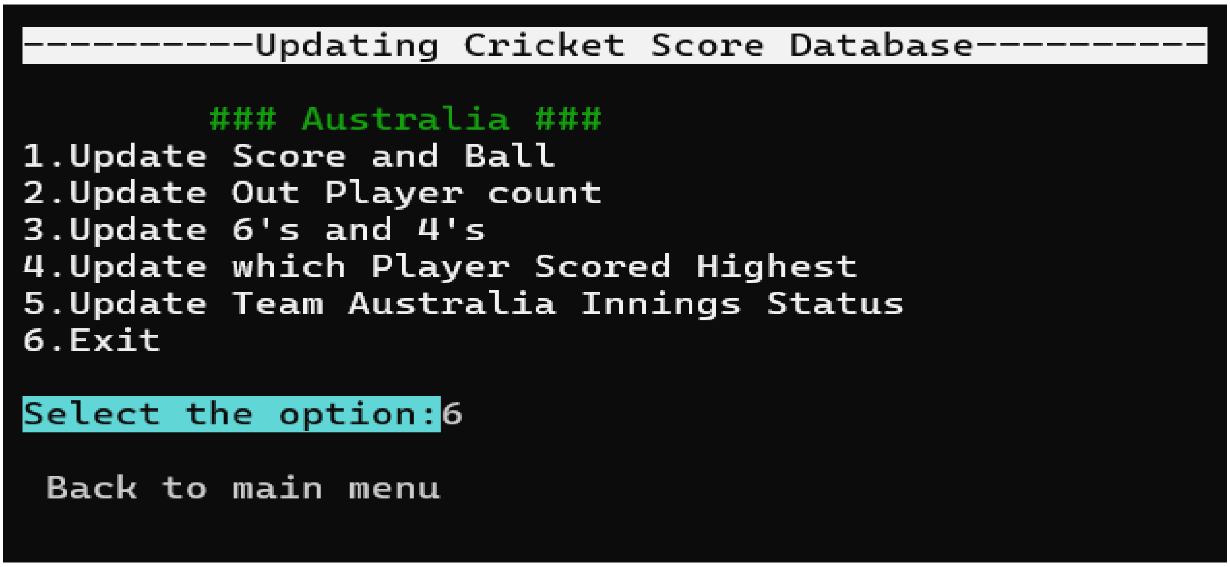
```
Select the option:4

        ### Australia ###

Enter the name of the Player which Highest Run Scorer on behalf of Team Australia
Name:David Warner
```

- **Option 5:** Choose the current inning scenario



```
Select the option:5

        ### Australia ###

 Is this Inning completed :
         1. YES
         2. No or Ongoing

Select the option:1
```

- **Option 6:** Exit the current sub-menu

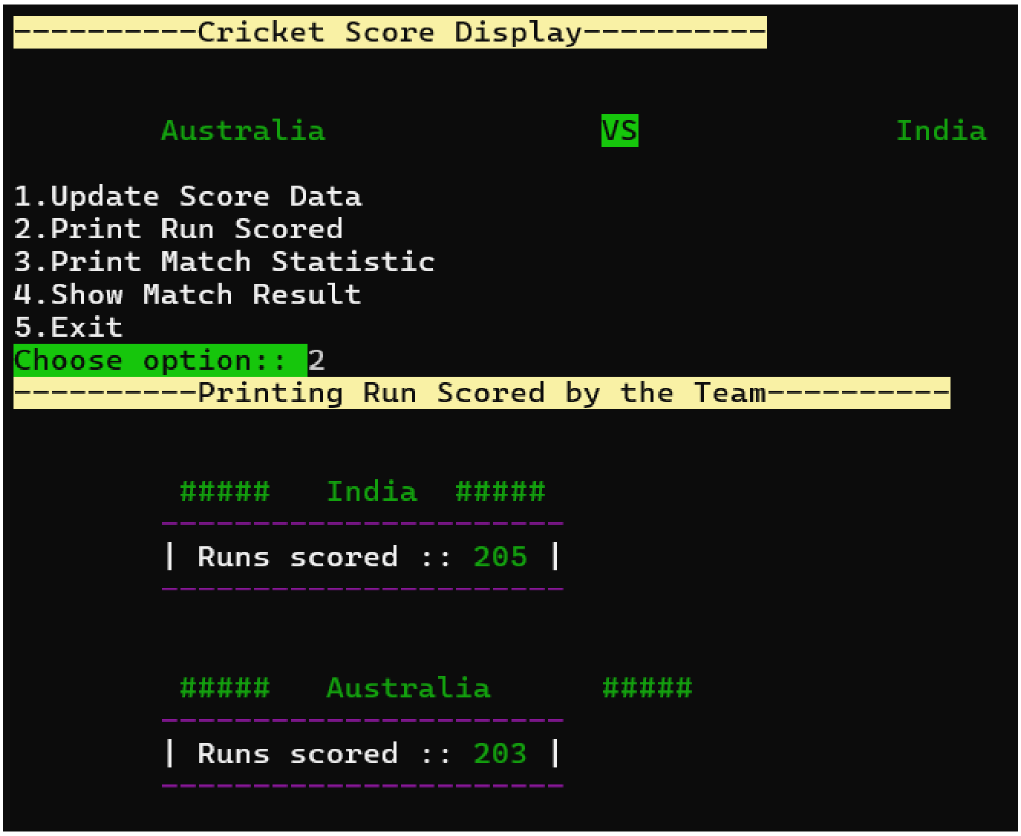


```
----------Updating Cricket Score Database----------

        ### Australia ###
1.Update Score and Ball
2.Update Out Player count
3.Update 6's and 4's
4.Update which Player Scored Highest
5.Update Team Australia Innings Status
6.Exit

Select the option:6

 Back to main menu
```

❖ **Display Runs and Statistics:**

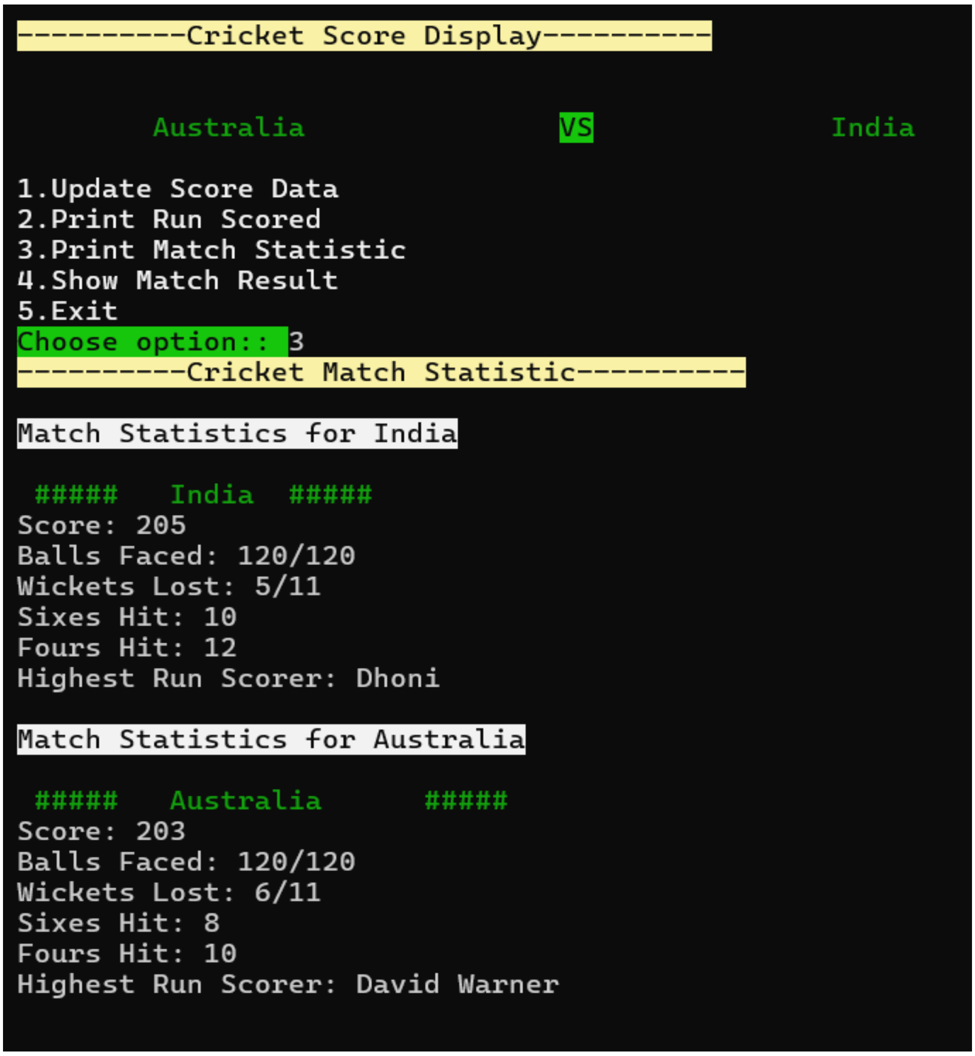
- Users can view the runs scored by each team and detailed match statistics at any time.

- **Option 2 :** Print Run Scored



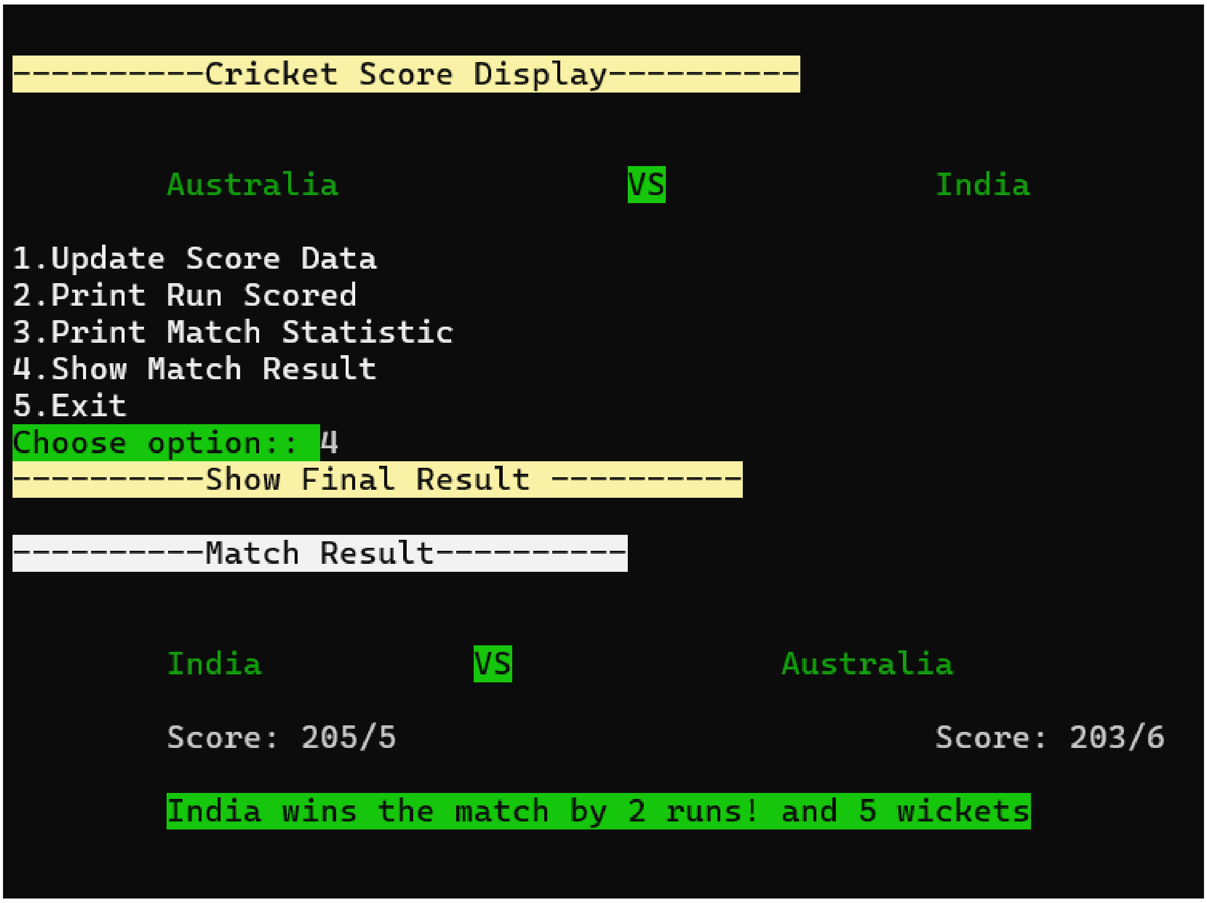
```
----------Cricket Score Display----------


        Australia              VS              India

1.Update Score Data
2.Print Run Scored
3.Print Match Statistic
4.Show Match Result
5.Exit
Choose option:: 2
----------Printing Run Scored by the Team----------


         #####   India  #####
        ----------------------
        | Runs scored :: 205 |
        ----------------------


         #####   Australia       #####
        ----------------------
        | Runs scored :: 203 |
        ----------------------
```

- **Option 3 :** Print Match Statistic for each team



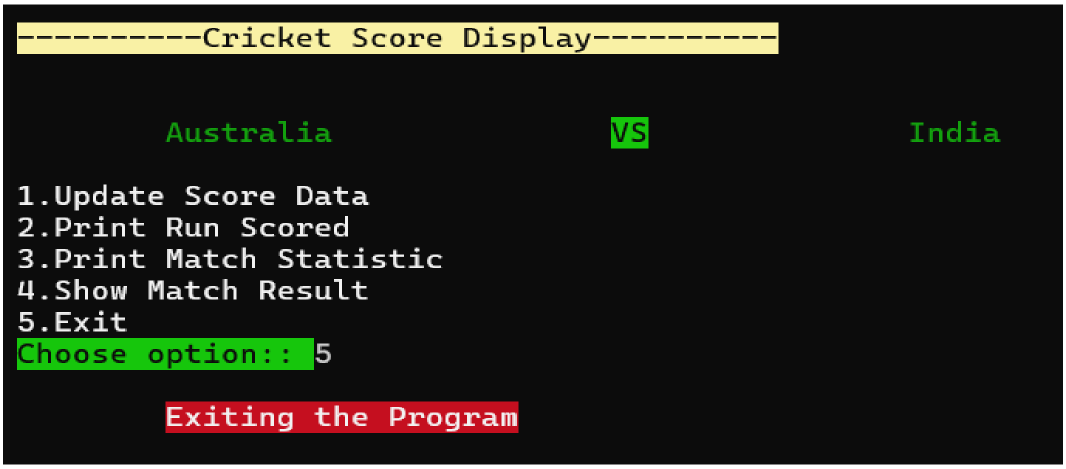
```
----------Cricket Score Display----------


        Australia              VS              India

1.Update Score Data
2.Print Run Scored
3.Print Match Statistic
4.Show Match Result
5.Exit
Choose option:: 3
----------Cricket Match Statistic----------

Match Statistics for India

 #####   India  #####
Score: 205
Balls Faced: 120/120
Wickets Lost: 5/11
Sixes Hit: 10
Fours Hit: 12
Highest Run Scorer: Dhoni

Match Statistics for Australia

 #####   Australia      #####
Score: 203
Balls Faced: 120/120
Wickets Lost: 6/11
Sixes Hit: 8
Fours Hit: 10
Highest Run Scorer: David Warner
```

- ❖ **Match Result:**
  - The program calculates and displays the final result of the match, indicating which team won or if the match is ongoing or tied.
  - **Option 4:** show Match Result



```
----------Cricket Score Display----------


        Australia                VS              India

1.Update Score Data
2.Print Run Scored
3.Print Match Statistic
4.Show Match Result
5.Exit
Choose option:: 4
----------Show Final Result ----------

----------Match Result----------


        India          VS              Australia

        Score: 205/5                           Score: 203/6

        India wins the match by 2 runs! and 5 wickets
```

- ❖ **Exit Program:**
  - **Option 5:** in main-menu for exit the program



```
----------Cricket Score Display----------


        Australia                VS              India

1.Update Score Data
2.Print Run Scored
3.Print Match Statistic
4.Show Match Result
5.Exit
Choose option:: 5

        Exiting the Program
```

9. Conclusion

The **Cricket Score Display Program** successfully achieves its objectives by providing a robust platform for managing cricket match scores and statistics. It provides a comprehensive and interactive way to manage and display cricket match scores and statistics. It combines functionality with user-friendly design, ensuring that users can interact with the program efficiently. By handling real-time score updates and presenting detailed match information, the program serves as a valuable tool for cricket fans and can be a foundation for more advanced applications.

10. Future Enhancements

- ❖ **Data Persistence**:
  - Implement file handling to save and load match data.
- ❖ **Graphical User Interface (GUI)**:
  - Develop a GUI version of the application for better user interaction.
- ❖ **Advanced Features**:
  - Include support for different match formats and additional cricketing rules.

❖ **Network Capabilities**:

- Enable real-time score updates over a network, allowing multiple users to access the scorecard.

❖ **Comprehensive Statistics**:

- Provide detailed analytics such as player form guides, historical performance, and predictive modelling.

❖ **Internationalization:**

- Support multiple languages to cater to a global audience.