DATA STRUCTURE ALGORITHM & APPLICATION

CT-159

PROJECT NAME: TOWER OF CARDS

FOR

Second Year

Computer Science

BATCH: 2023-24

GROUP MEMBERS:

1. **HUZAIFA IMTAZ, CT-096**
2. **MUHAMMAD ALI, CT-074**
3. **MUHAMMAD HASSAN SIDDIQUI, CT-094**
4. **MUHAMMAD KHUBAIB, CT-097**

DISCIPLINE: FSCS (COMPUTER SCIENCE)

SECTION: B

SEMESTER: 3rd , FALL

DEPARTMENT OF CSIT

NED UNIVERSITY OF ENGINEERING & TECHNOLOGY, KARACHI, PAKISTAN.

TABLE OF CONTENTS:

Introduction

The "Tower of Cards" project is a Console game-based implementation of a card game with strategic gameplay. The game will involve power cards that introduce dynamism effects during gameplay, enhancing the interactive nature of the game. The main goal shall be to gain points playing cards while using power cards for strategic gains.

Problem Statement

In the absence of a dynamic, strategy-based card game simulation, the "Tower of Cards" project aims to fill this gap by offering an engaging platform for users to enjoy a game based on decision-making and chance.

Features

Card Initialization: Deck creation and shuffling with an even distribution of power cards.

Player Turn Handling: Players take turns selecting cards, with the option to use power cards for strategic effects.

Power Card Effects: Cards such as Ace, King, Queen, and Joker come with special effects like more turns or doubling points.

Point scoring: Players get points when they correctly play cards.

Protected Tower: Some cards protect the player from penalty for a certain number of rounds.

Game End: The game comes to its end after a specific number of rounds or when all cards are played.

Implementation Details

1. Card Initialization and Shuffling

Deck Generation: **createDeck()** The method initializes the deck of 52 cards which consists of the four suits (Hearts, Diamonds, Clubs, Spades) and the 13 values: 2-10, J, Q, K, A

Shuffle: The **shuffleDeck()** function utilizes a simple random number generator based on the system time to scramble the cards.

2. Color Coding for Cards

Color Assignment: The **getCardColor()** method assigns a particular ANSI color code to each card according to its suit:

Hearts: Red

Diamonds: Light Blue

Clubs: Yellow

Spades: Light Green

Power Cards (J, Q, K, A): Purple

Display Functionality: The **displayCard()** and **displayHand()** methods make use of these color codes to differentiate cards visually when printed to the console.

3. Sound Effects

Power Card Effects: A beep sound is generated at the time of play when a power card is used, which enriches the user experience.

4. Gameplay Mechanics

1. Turn of Player and Computer
2. Players draw cards from their hand and score points based on the value of the drawn card.
3. The computer draws cards from the deck in its turn.
4. Power Card Effects: Specific effects are added at the time of play of the power cards:
5. J (Joker): Two more cards are drawn.
6. Q: Save the tower for next three turns.
7. K: Subtract the opponent's points.
8. A: Multiply the value for the next three cards by two.

5. Scoring

The game calculates the score based on card values.

Number cards (2-10): Value on card

Power cards: Extra points for each power card; J=11, Q=12, K=13, A=15.

Division of Tasks :

Muhammad Ali (CT-074) :

- Developed the main game logic, including the round-based structure and turn handling for players.
- Managed the implementation of power card effects and scoring mechanisms.

Huzaifa Imtiaz (CT-096) :

- Implemented the createDeck() and shuffleDeck() functions to generate and randomize the card deck.
- Designed the Card struct for representing individual cards.

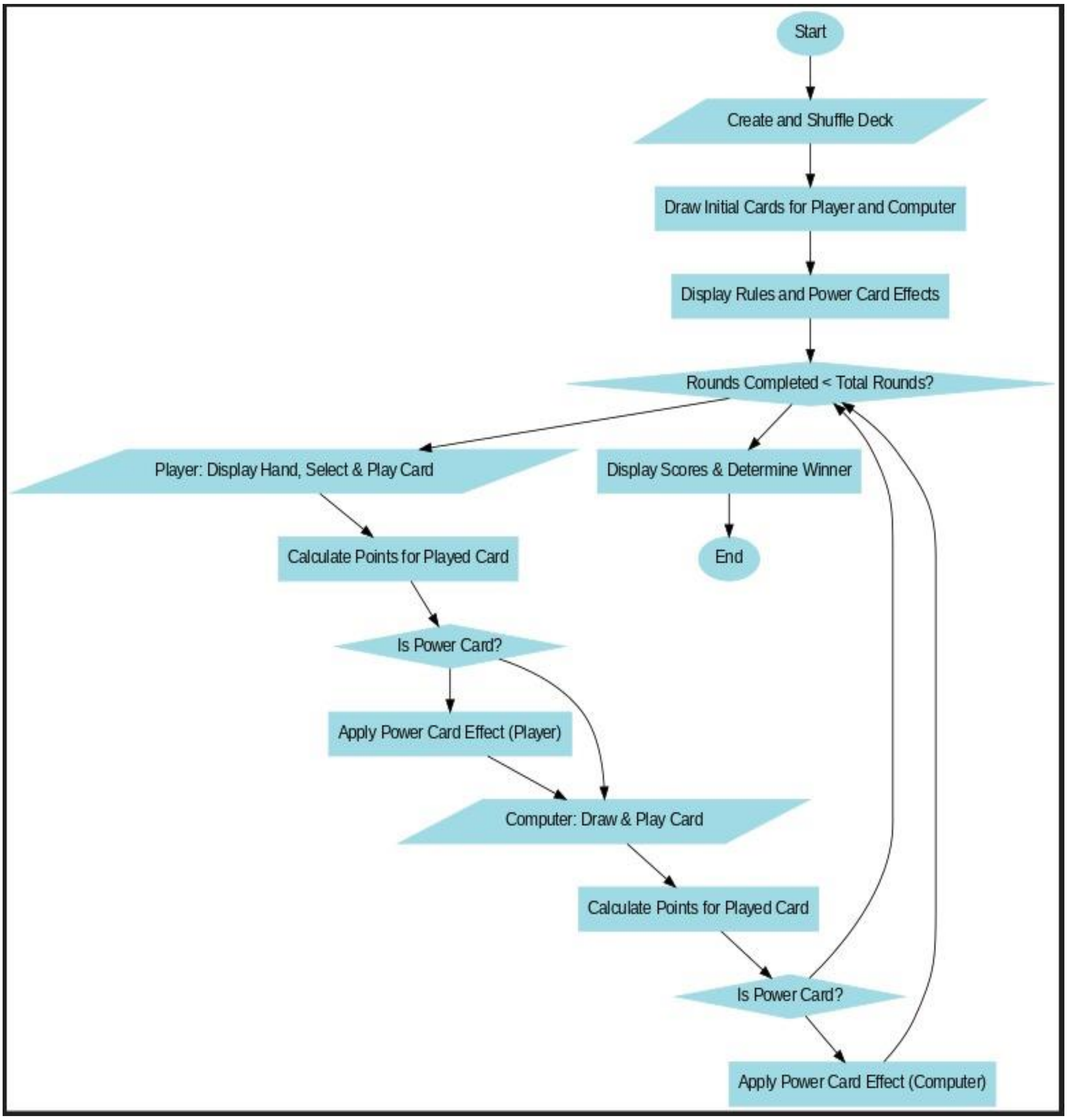
Muhammad Hassan Siddiqui (CT-094) :

- Worked on user experience elements such as ANSI color-coded card displays (getCardColor(), displayCard()).
- Added sound effects to enhance gameplay immersion.

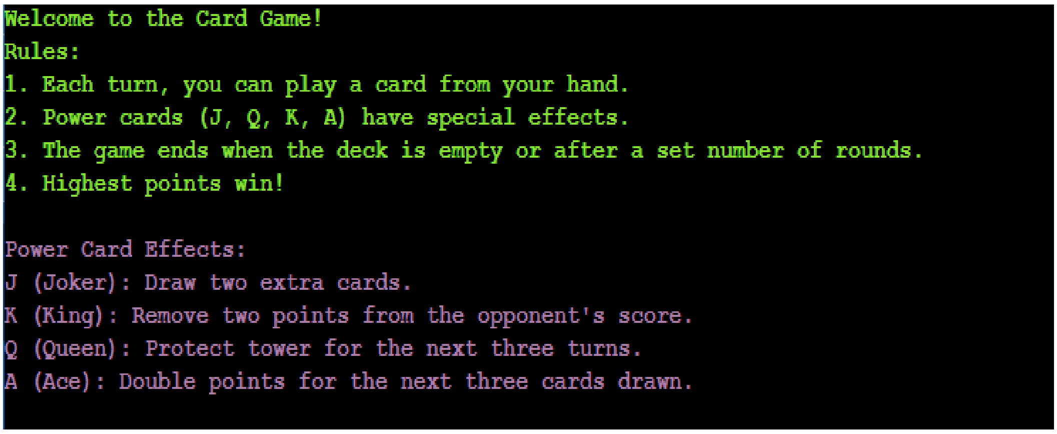
Muhammad Khubaib (CT-097) :

- Designed the report generation system (generateReport()), which provides a detailed summary of rounds and scores.
- Prepared the project documentation, including the final report.

FLOW CHART:


Start
Create and Shuffle Deck
Draw Initial Cards for Player and Computer
Display Rules and Power Card Effects
Rounds Completed < Total Rounds?
Player: Display Hand, Select & Play Card
Display Scores & Determine Winner
End
Calculate Points for Played Card
Is Power Card?
Apply Power Card Effect (Player)
Computer: Draw & Play Card
Calculate Points for Played Card
Is Power Card?
Apply Power Card Effect (Computer)

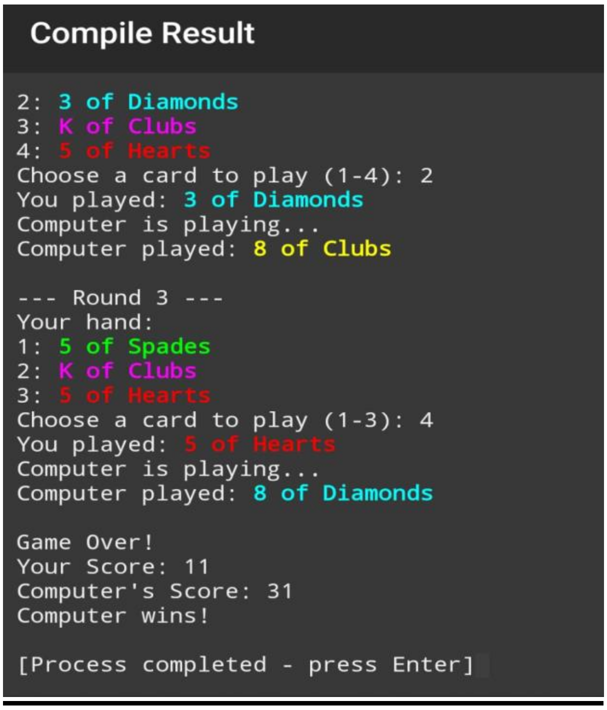
GAME RULES:



```
Welcome to the Card Game!
Rules:
1. Each turn, you can play a card from your hand.
2. Power cards (J, Q, K, A) have special effects.
3. The game ends when the deck is empty or after a set number of rounds.
4. Highest points win!

Power Card Effects:
J (Joker): Draw two extra cards.
K (King): Remove two points from the opponent's score.
Q (Queen): Protect tower for the next three turns.
A (Ace): Double points for the next three cards drawn.
```

OUTPUT



```
2: 3 of Diamonds
3: K of Clubs
4: 5 of Hearts
Choose a card to play (1-4): 2
You played: 3 of Diamonds
Computer is playing...
Computer played: 8 of Clubs

--- Round 3 ---
Your hand:
1: 5 of Spades
2: K of Clubs
3: 5 of Hearts
Choose a card to play (1-3): 4
You played: 5 of Hearts
Computer is playing...
Computer played: 8 of Diamonds

Game Over!
Your Score: 11
Computer's Score: 31
Computer wins!

[Process completed - press Enter]
```

Limitations/Future Enhancements

Limitations:

- Console-based interface limits interactivity and graphical presentation.
- Limited error handling for unexpected user inputs.

Future Enhancements:

- Introduce a graphical user interface for enhanced user experience.
- Add multiplayer capabilities.
- Implement advanced strategies for computer opponents.

Conclusion

The "Tower of Cards" project demonstrates a strategic and dynamic card game simulation, showcasing proficiency in C++ programming concepts such as arrays, functions, stacks and conditional logic. Future enhancements can further expand its scope and interactivity.

The End