Project Information: BotBreak - Call Break Game Bot

Project Overview

BotBreak is a backend programme designed to simulate a Call Break card game with modified rules. This bot autonomously plays the game, making strategic decisions based on the current game scenario.

Key Features

1. **Autonomous Bot Gameplay**: BotBreak is designed to play the Call Break game autonomously, selecting and playing cards according to different scenarios.
2. **Strategic Logic Functions**: The bot's decision-making logic is implemented in functions from func1() to func7(). The process begins with func1(), which, based on its decisions, calls the next relevant function until a final decision is made by a leaf node function.
3. **Efficient Game Management**: The game uses various data structures such as linked lists, queues, and arrays to manage players, cards, and game flow efficiently.

Special Logic Strategies and Efficient Management

The core logic of the bot is implemented in functions from func1() to func7(). The logic follows a special strategy for decision-making:

1. **Starting Point**: The process begins with func1(), which evaluates the initial game conditions.
2. **Decision Tree**: Based on the evaluations made in func1(), the function calls the next relevant function in the sequence to further narrow down decisions.
3. **Final Decision**: This chain of function calls continues until a leaf node function is called, where the final decision is made on which card to play.

This hierarchical decision-making strategy ensures that the bot can handle various game scenarios effectively, making informed decisions at each step. Powered by these special logic strategies and efficient management, the bot can dynamically adapt to different game conditions, and optimize its moves.

Key Functions and Their Efficiency

- **nextPlayer(wonPlayer)**: Determines the next player based on the Players linked list. This function efficiently manages the circular queue of players. Dynamically selecting the next player keeps the gameplay continuous.

- **distribute()**: Randomly distributes cards among the four players from the CardList linked list. This function ensures an even and fair distribution of cards, setting up the game for balanced play.
- **initialize()**: Prepares the game for play by initializing the CardInfo array, filling the players, populating the card list, distributing cards, and sorting the cards for each player. This comprehensive setup function ensures that the game starts in a well-organized state. By automating the setup process, it reduces manual intervention and potential errors.
- **control()**: Starts the game and manages its progression. This central function oversees the game's flow, coordinating the actions of other functions and ensuring that the game runs smoothly from start to finish. Efficient control of the game flow ensures that all game rules are followed, and the game logic is executed correctly.

Summary

BotBreak leverages strategic logic and efficient game management to simulate an autonomous Call Break game bot. The programme runs entirely in the terminal and uses a hierarchical decision-making process to handle various game scenarios. This robust solution provides intelligent and dynamic Call Break gameplay, ensuring fair play and efficient game management.

Logic Functions Implementation and Block Diagram