🏆KOPER🏆
ALGO-STRATEGIC POKER TOURNAMENT DOCUMENTATION

This document outlines the rules, structure, and technical specifications of the poker match engine, based on the provided environment files (engine_core.py, dummy_strategies.py, main.py).

1. MATCH AND GAME STRUCTURE

The tournament follows a modified, cap-limited No-Limit Texas Hold'em format played over multiple hands (games).

Parameter	**Value (from engine_core.py)**	**Description**
Total Games (Hands)	NUM_GAMES = 50	The total number of hands played in a single match.
Players per Match	5	The number of participants in the match.
Starting Stack	STARTING_STACK = 10000$	The initial money each player begins the match with.
Mandatory Buy-in	BUY_IN = 100$	The fee paid by every active player to enter each new game.
Game Deck	Standard 52-card deck	Cards are represented as 'RankSuit', e.g., 'As' (Ace of Spades).

2. PLAYER STATE AND ELIMINATION

Match Elimination

- A player is permanently eliminated from the entire **match** (set to is_lost_match = True) if they cannot afford the mandatory BUY_IN = 100$ at the start of a new game (Round 0).
- Any remaining money is forfeit to the pot.

Game Elimination (Fold)

- A player can only choose to “**FOLD”** in **Round 1**.
- A player who folds forfeits all money already placed in the pot (the BUY_IN = 100$ and their Round 1 bet).
- The player is out of the current game but remains in the match for subsequent games.

3. GAME FLOW AND BETTING ROUNDS

A single game consists of one initial buy-in round and three distinct betting rounds, followed by a Showdown.

3.0. Round 0: Setup and Buy-in

- **Buy-in Action:** Each active player pays the BUY_IN = 100$ from their stack into the pot.
- **Dealing:** Each active player receives 2 **Hole Cards**. A set of 5 **Community Cards** is drawn but remains unseen.
- **Elimination Check:** Players must have a minimum stack of $100. If not, they are eliminated from the match.

3.1. Round 1: Flop Betting (The Bet Decision)

- **Cards Visible:** The first **3 Community Cards** (The Flop) are revealed.
- **Minimum Stack Check:** Players must have a minimum stack of *100$* at the start of Round 1. If not, they are eliminated from the match.
- **Player Action (`strategy.round1`):**
  - **Action Type:** Must choose between `("fold", 0)` or `("play", amount)`.
  - Bet Range: If playing, the Round_1_Bet must be in the range:
    $100.00 ≤ Round_1_Bet ≤ $300.00
  - The bet is immediately subtracted from the player's stack and added to the pot.
- **Early Win:** If only one player remains active (all others fold), that player wins the pot immediately.
- **NOTE:** This is the last (and the only) round where you or your opponents can fold.

3.2. Round 2: Turn Betting (The Bet Adjustment)

- **Cards Visible:** The **4th Community Card** (The Turn) is revealed.
- **Player Action (`strategy.round2`):**
  - **FOLDING IS NOT ALLOWED.**
  - Bet Range: The new Round_2_Bet must be constrained by the previous Round_1_Bet, i.e.,
    0.5 x Round_1_Bet ≤ Round_2_Bet ≤ 1.5 x Round_1_Bet
  - The bet must not exceed the player's current stack.

3.3. Round 3: River Betting (The Final Bet)

- **Cards Visible:** The **5th and final Community Card** (The River) is revealed.
- **Player Action (`strategy.round3`):**
  - **Folding is not allowed.**
  - Bet Range: The final Round 3 Bet must be constrained by the previous Round 2 Bet i.e.,
    0.75 x Round_2_Bet ≤ Round_3_Bet ≤ 1.25 x Round_2_Bet
  - The bet must not exceed the player's current stack.

Input Variable	Description	Source
hole_cards	Your 2 private cards.	Not useful as win_prob will be given.
visible_community	The 3 or 4 or 5 visible Community Cards.	Not useful as win_prob will be given.
r1_bets	An array of all players' Round 1 final bet amounts. (player index = array index)	Will be available to players/strategies in round 2 and 3.
r2_bets	A dictionary of all players' Round 2 final bet amounts. (player index = array index)	Will be available to strategies in round 3.
current_stacks	The current stack value of all players. (player index = array index)	Might be useful, given in all the rounds.
pot	The total money in the pot.	Useful in Round 3 if you have a strong hand, since winnings are capped by your bet size (which is bounded).
win_prob	Your calculated probability of winning the hand (equity), given your cards.	In round 1, 2, 3, it will be given in each round (different).

NOTE: win_prob **is not the probability of the player winning the given game, but it is the probability of the player winning a poker game given his cards.**

For example, if all the players have good cards, all of them will have high win_prob.

4. SHOWDOWN AND PAYOUT LOGIC

The game follows the same rule as Poker to decide the winner. However, you do not need to know them because the chances of winning are

4.1. Hand Evaluation

- The winner is determined by the best 5-card combination from the 7 available cards (2 Hole Cards + 5 Community Cards). A **lower numerical score indicates a better hand**.

4.2. Pot Allocation and Winning Cap (The Key Rule)

The entire pot is **not** automatically awarded to the winner.

1. Player's Total Contribution:
   PTC = BUY_IN + Round_1_Bet + Round_2_Bet + Round_3_Bet.
2. Winning Cap: The maximum amount a winning player(s) can receive is capped at four times their total contribution:
   WC = 4 × players_total_contribution
3. Actual Payout: Each winner(s) receives their share of the pot, limited by the cap:
   Actual Win = min(Total Pot, Winning Cap)
4. **Redistribution Law:** Any money remaining in the pot after the winners have been paid (the "Remaining Pot") is **redistributed equally** among all players who were active in Round 2 and 3 (i.e., did not fold in Round 1).

5. MATCH HISTORY ACCESS

In real Poker, players adapt or update their strategies observing other players from past games, the Koper environment also allows strategies to update themselves by accessing match history.

To support adaptive strategies, the engine provides access to the **match history** of all previously completed games.

At the start of each new game, strategies receive a structured log of past games via the `initialize_game()` method. This allows strategies to analyze opponent behavior across the match and adjust their decisions based on observed long-term patterns, rather than treating each game independently.

Only information from **completed games** is included. No hidden cards or future game state is exposed.

5.1. Match History Contents

Each history entry corresponds to one completed game and includes the following:

Item	Description
Player Index	Unique identifier for each player
Hole Cards	Private cards dealt to each player
Community Cards	Cards revealed during the game
Fold Status	Whether the player folded in Round 1
Round Bets	Final bet amounts for Rounds 1–3
Win Probabilities	`win_prob` values across rounds
Final Hand Score	Hand strength at showdown
Final Pot Size	Total Pot Value
Final Stack	Player’s stack after the game