



# QUANT PROBLEM STATEMENT

POKER  
PERSPECTIVE



FINOPOLY





# The Build-Up

Quantitative finance and poker share a deep, fundamental connection. Both are games of incomplete information, where the best participants consistently make superior decisions under uncertainty. Success is not defined by the outcome of a single hand or a single trade, but by a long-term strategy that maximizes expected value (EV).

In this challenge, you will step into the shoes of a quant. You will:

- Model uncertainty and estimate probabilities.
- Calculate the expected value of your actions.
- Manage risk versus reward.
- Develop a strategy that adapts to your opponent's behavior.

Your task is to build a bot that plays a simplified version of poker. No prior poker or extensive coding experience is required; we are interested in your logical reasoning and quantitative approach to problem-solving.

# The Challenge

You are to design and build a “Quant-Inspired Poker Bot”. This bot will compete in a tournament against bots created by other applicants. Your bot must programmatically decide on one of three actions—Fold, Call, or Raise—based on the game state provided each round.





# Game Rules: Simplified 3-Card Poker

Each match consists of many independent rounds. The gameplay for a single round is as follows:

## A. The Deal

1. The game uses a standard 52-card deck, shuffled before each round.
2. Each of the two players is dealt two private cards (your "hole cards").
3. One single community card is dealt face-up on the "table".

## B. Hand Composition

A player's hand is always composed of their two hole cards plus the one community card, for a total of three cards.

## C. Hand Rankings & Tie-Breakers

The hand rankings are listed below from highest to lowest. Tie-breakers are crucial and are applied in sequence.

**1. Straight Flush:** Three cards of the same suit in sequence.

- **Tie-Breaker:** The hand with the higher-ranking top card wins. (e.g., 6-7-8 beats 3-4-5). The Ace can be used for Q-K-A (high) or A-2-3 (low). The A-2-3 straight is the lowest-ranking straight.

**2. Three of a Kind (Trips):** Three cards of the same rank.

- **Tie-Breaker:** The hand with the higher-ranking three cards wins. (e.g., 7-7-7 beats 5-5-5).

**3. Straight:** Three cards of sequential rank, not all of the same suit.

- **Tie-Breaker:** The hand with the higher-ranking top card wins. (e.g., 8-9-T beats 4-5-6).





#### 4. Flush: Three cards of the same suit, not in sequence.

- **Tie-Breaker:** The hand with the single highest-ranking card wins. If the highest cards are tied, the second-highest cards are compared. If those are also tied, the third-highest cards are compared. (e.g., K♦ 8♦ 3♦ beats Q♦ J♦ 9♦ because the King beats the Queen).

#### 5. Pair: Two cards of the same rank, and one card of a different rank (the "kicker").

- **Tie-Breaker (in order):**

- i. The hand with the higher-ranking pair wins. (e.g., 9-9-5 beats 7-7-A).
- ii. If both hands have the same pair, the hand with the higher-ranking kicker wins. (e.g., 9-9-K beats 9-9-Q).

#### 6. High Card: Three cards that do not form any of the above hands.

- **Tie-Breaker:** Works identically to the Flush tie-breaker; the highest card wins, followed by the second-highest, then the third.
- If after all tie-breaker rules are applied the hands are still identical, the round is a Tie.

### D. Actions and Scoring

- After the cards are dealt, both players simultaneously choose an action. The points awarded for the round are determined by the combination of actions and the showdown result, as shown in the table below.





Player 1 Action	Player 2 Action	Outcome Description	P1 Points	P2 Points
FOLD	FOLD	Both Players Forfeit	0	0
CALL	FOLD	P2 forfeits, P1 wins the round.	+2	-1
FOLD	CALL	P1 forfeits, P2 wins the round.	-1	+2
Raise	FOLD	P2 forfeits, P1 wins the round.	+3	-1
FOLD	Raise	P1 forfeits, P2 wins the round.	-1	+3
CALL	CALL	Showdown, P1 wins	+2	-2
CALL	CALL	Showdown, P2 wins	-2	+2
Raise	CALL	Showdown, P1 wins	+3	-2
Raise	CALL	Showdown, P2 wins	-3	+2
CALL	Raise	Showdown, P2 wins	-2	+3
CALL	Raise	Showdown, P1 wins	+2	-3
Raise	Raise	High-Risk Showdown. P1 wins, gets bonus.	+3	-3
Raise	Raise	High-Risk Showdown. P2 wins, gets bonus.	-3	+3
Any Showdown	Any Showdown	Hands are identical.	0	0





**To summarize:** a standard showdown (Call vs. Call) is a +/-2 point exchange, but Raising increases your potential winnings and losses to +/-3 points. Folding when your opponent stays in results in an automatic -1 point penalty for you, awarding points to them.

## The Quant Angle: How You Will Be Evaluated

Your submission will be judged on four key pillars. The justification of your strategy is as important as its performance.

- 1. Bot Performance (Win Rate):** Your bot's overall score and win rate across the tournament.
- 2. Quantitative Strategy:** The logic and mathematical reasoning behind your bot's decisions. This is assessed primarily through your one-page explanation. We are looking for evidence of:
  - Probability Estimation (e.g., calculating odds of making a pair, winning a showdown).
  - Expected Value (EV) Logic (e.g., a model for the EV of Calling vs. Raising).
  - Adaptive Strategy (e.g., detecting and exploiting opponent tendencies).
- 3. Stability:** Your bot must run without crashing.

## Technical Task & Submission

### A. Your Task

You will be provided with a Python script, player\_template.py. This file contains all the necessary code to run the game.

- Your ONLY task is to modify the 'decide\_action' function within this file.
- The template includes helper functions and comments to guide you. You are free to modify or add to the code as you see fit to implement your strategy.





## B. What to Submit

Please submit two files:

1. Your Bot Script: Rename your modified player\_template.py file to team\_leader\_roll\_no.py (e.g., 250101023.py).
2. Your One-Page Explanation: A PDF document outlining your strategy, as described in the next section.

**Submission Link:** ——[\*\*Click here\*\*](#)

### The One-Page Explanation

This document is your opportunity to demonstrate your quantitative thinking. It must concisely cover the following points:

- Probability Estimation: How did you model the game? How did you calculate the probability of forming certain hands or winning a showdown?
- Strategy & Logic: What is your bot's core decision-making logic? How does it decide whether to Fold, Call, or Raise?
- Assumptions: What assumptions did you make about the game or your opponents?
- Risk Modelling: How did you approach the "Raise" action? Explain your logic for when to risk escalating the stakes for a higher reward.
- The "Quantitative" Element: Briefly explain what makes your bot's strategy "quantitative."

**A word of advice:** Relying on AI for your entire strategy will likely produce a generic solution similar to many others, while the most successful submissions will demonstrate unique and independent reasoning distinct from others.





## Guidelines

- The teams must consist of **3-4 members**.
- The deadline for submission is **22nd of December 2025**

## Resources and working code template:-

- Python for Beginners - Learn Coding with Python in 1 Hour  
**(Programming Resource to get familiar with Python)**
- Math of Poker - Basics | Brilliant Math & Science Wiki **(A Primer on Probability with emphasis on Poker - Remember we are dealing with a simplified 3 card version of the game)**
- Expected Value | Poker Mathematics
- MIT 15.S50 Poker Theory and Analysis, IAP 2015 - YouTube  
**(Super Extensive - Not Recommended)**
- <https://bicyclecards.com/how-to-play/basics-of-poker> **(Basics of poker)**

Code template to be used → [Click Here](#)

