Simulation of a Coin-Flipping Game

Problem description

Objective:

The goal of this project is to determine the expected profitability of participating in a specific coin-flipping game. This game continues until the difference between the number of heads and tails flipped reaches three. Each flip of the coin costs the player ¥1, and upon the game's conclusion—when the head-tail difference is exactly three—the player receives a reward of ¥8.

Rules:

1. The coin used for flipping is unbiased, meaning it has an equal probability (1/2) of landing on either heads or tails.
2. The player cannot opt out once the game has started. They must continue flipping the coin until the head-tail difference is three.
3. The cost to flip the coin is ¥1 per flip.
4. The reward for completing the game is ¥8, granted only when the difference between the number of heads and tails is exactly three.
5. The game's profitability is determined by the number of flips: if the game concludes in fewer than eight flips, the player profits; if it requires more than eight flips, the player incurs a loss.

Approach:

To assess whether the game is worth playing, a simulation will be conducted. The simulation will replicate the game's mechanics by generating a sequence of random outcomes for the coin flips, tracking the difference between heads and tails, and calculating the net gain or loss for each game iteration. This method allows for a statistical analysis of the game's expected value without actual financial risk.

Questions:

- What is the average number of flips required to end the game?
- Is the expected value of playing the game positive, indicating profitability?
- How does the probability distribution of the game's outcomes influence decision-making?

Model

Mathematical Model:

1. State Variable $M(n)$: Let n denote the number of coin flips. $M(n)$ represents the state of the system, defined as the difference between the number of heads and tails after n flips. This variable captures the dynamic behavior of the coin-flipping process.

$$M(n) = Number\ of\ heads - Number\ of\ tails \tag{1}$$

2. System State Transition Rules:

- If the result of the n^{th} coin flip is heads, then the state variable is updated as follows:

$$M(n) = M(n-1) + 1 \tag{2}$$

- If the result of the n^{th} coin flip is tails, then:

$$M(n) = M(n-1) - 1 \tag{3}$$

3. Termination Condition: The simulation ends when the absolute value of $M(n)$ reaches 3, i.e., $|M(n)| = 3$. This condition signifies the achievement of the game's objective, resulting in the conclusion of a game round.
4. Economic Model (Earnings E): The participant's earnings from a game are calculated as the fixed reward minus the cost incurred from each flip. With a reward of ¥8 and a cost of ¥1 per flip, the earnings are given by:

$$E = 8 - n \tag{4}$$

where n is the total number of flips required to end the game.
5. Average Metrics:
- Average Number of Flips $\bar{n}$: The average number of flips required to meet the game's termination condition across all simulated games is calculated as:

$$\bar{n} = \frac{\sum_{i=1}^{m} n_i}{m} \tag{5}$$

where m is the total number of simulated games, and n_i is the number of flips in the i^{th} game.
- Average Earnings $\bar{E}$: The average earnings across all simulations are calculated as:

$$\bar{E} = \frac{\sum_{i=1}^{m} E_i}{m} \tag{6}$$

Where $E_i = 8 - n_i$ represents the earnings from the i^{th} game.

The aim of this mathematical model is to utilize computational simulations to evaluate the statistical properties and expected profitability of participating in the coin-flipping game under defined conditions.

Program Model

This program model is designed to simulate the dynamics of a coin-flipping game within a graphical user interface (GUI) application developed in Visual Studio C++. The simulation assesses the profitability of the game, where the objective is to continue flipping an unbiased coin until the difference between the number of heads and tails reaches a predefined condition.

Model Components:

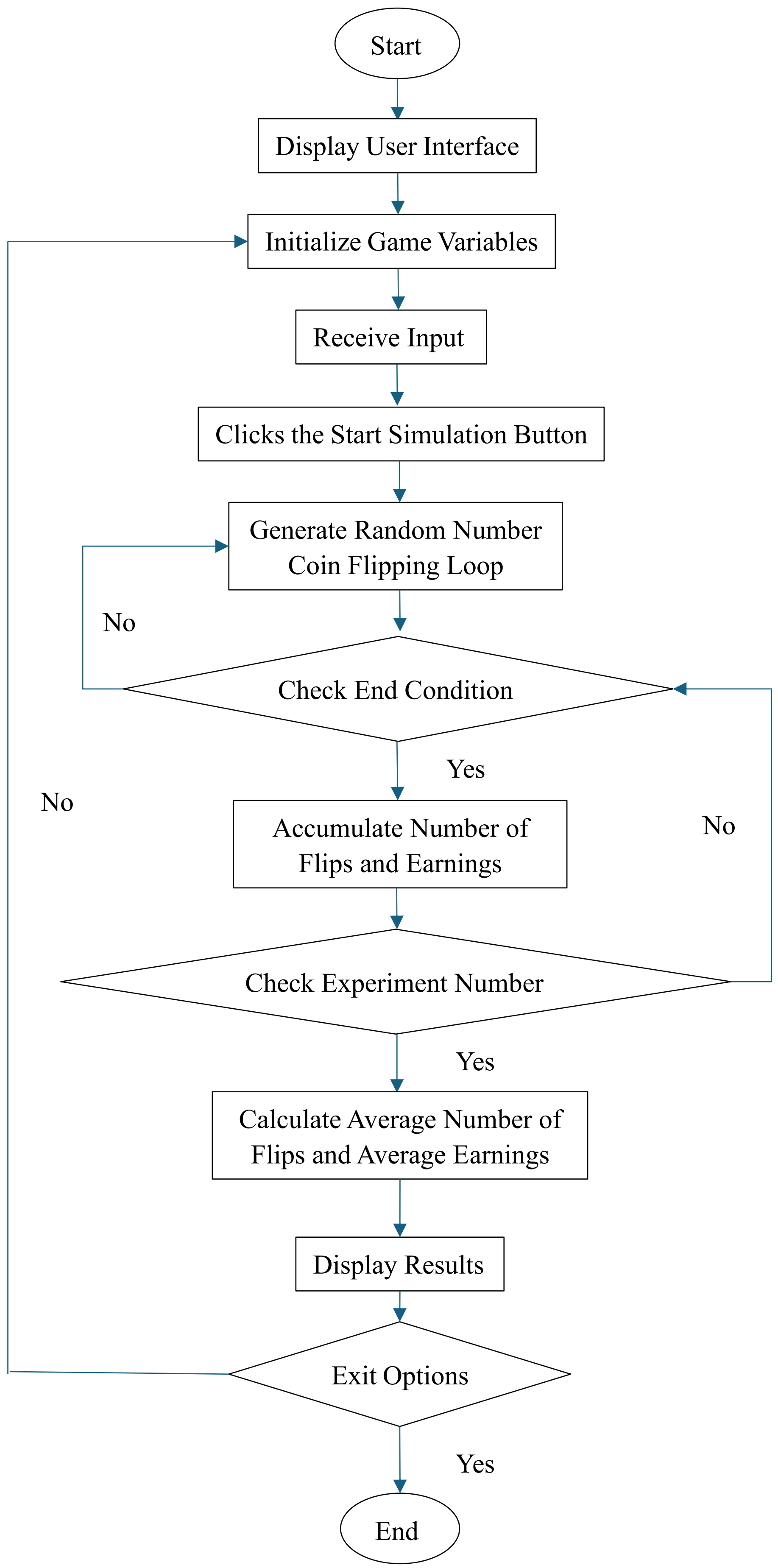
1. Graphical User Interface (GUI): Provides the user with interactive elements to input simulation parameters such as cost per round, reward per game, condition for game completion, and number of experiments (simulations) to run.
2. Simulation Parameters:
   - `*m_costPerRound*`: The cost of each coin flip.
   - `*m_rewardPer*`: The reward for achieving the condition, set to 8 as per game rules.
   - `*m_condition*`: The difference in the count of heads and tails needed to conclude a game.
   - `*m_experiment*`: The number of games to simulate.
3. Random Number Generator (RNG): Utilizes C++'s `rand()` function to simulate the outcome of each coin flip, with a 50% chance for heads or tails.
4. State Variable - M: Represents the current difference between the number of heads

and tails after `*n*` flips.
5. Simulation Logic: Executes within the `*OnBnClickedButton1()*` function, simulating each game by continuously flipping the coin until the absolute difference (`abs(M)`) reaches the `*m_condition*`. It calculates total flips and total earnings across all simulations to evaluate the game's profitability.
6. Results Calculation:
   - Calculates the average number of flips and average earnings across all simulated games.
   - Outputs these averages to the GUI for user analysis.
7. Reset Functionality: Offers the ability to clear all input and output fields via the `*OnBnClickedButton2()*` function, allowing for fresh simulations without restarting the application.

The primary aim is to provide a user-friendly platform for simulating and analyzing the coin-flipping game's outcomes, determining the average flips required and the overall profitability based on user-defined conditions and the number of simulations.

Users interact with the GUI to set simulation parameters and initiate the simulation process. The program then simulates the defined number of games, providing insights into the game's statistical behavior and potential profitability.

Program flowchart


Start
Display User Interface
Initialize Game Variables
Receive Input
Clicks the Start Simulation Button
Generate Random Number
Coin Flipping Loop
No
Check End Condition
Yes
No
Accumulate Number of
Flips and Earnings
No
Check Experiment Number
Yes
Calculate Average Number of
Flips and Average Earnings
Display Results
Exit Options
Yes
End

Outputs analysis

After conducting simulations of the coin-flipping game with varying numbers of trials (1,000; 10,000; and 100,000), the results offer an interesting perspective on how the game's outcome stabilizes with an increasing number of simulations:

- For 1,000 simulations, the average number of flips required was 9.22, with an average earnings of approximately -¥1.22.
- For 10,000 simulations, the average number of flips slightly decreased to 8.94, with an average earnings of approximately -¥0.94.
- For 100,000 simulations, the average number of flips was around 9.04, with an average earnings of approximately -¥1.04.

Deep Analysis of Results

The results indicate a pattern where the average number of flips tends to stabilize around 9 flips as the number of simulations increases. This suggests that, on average, it requires just over 9 flips for the difference between heads and tails to reach 3. This outcome is slightly more than the break-even point of 8 flips, which is why the average earnings tend to be negative.

The slight variations in average earnings across different simulation sizes could be attributed to the random nature of each simulation. However, as the number of simulations increases, these averages become more consistent and reliable, demonstrating the law of large numbers.

The consistent negative average earnings across all simulation scales reinforce the conclusion that the coin-flipping game, as defined, is statistically unfavorable for players over the long term. The fixed reward of ¥8 does not compensate for the average cost incurred due to the slightly higher-than-expected number of flips needed to meet the game's winning condition. This analysis highlights the importance of understanding probabilistic outcomes in games of chance and could serve as a caution against engaging in games with similar payout structures without fully understanding the expected outcomes.

Summary

This project conducted a detailed simulation and analysis of a coin-flipping game, where the objective is to continue flipping an unbiased coin until the difference between the number of heads and tails reaches three. Each flip costs ¥1, and upon achieving the objective, the player receives a reward of ¥8. Through simulations at various scales (1,000; 10,000; and 100,000 trials), the study aimed to assess the game's profitability and understand the statistical dynamics at play.

The core findings from the simulations are as follows:

- The average number of flips required to reach the game's objective stabilizes around 9, slightly above the break-even point of 8 flips.
- Consequently, the average earnings from playing the game tend to be negative, approximately -¥1.00 across various simulation scales, indicating a loss for the player over the long term.

These results suggest that the game, as structured, is statistically unfavorable for players, with the cost of participation often exceeding the fixed reward. This project highlights the importance of probabilistic analysis in assessing the fairness and potential profitability of games of chance.