The Big Bet: A Smart Contract Based Closed Lottery

For this final assignment you will create a contract to implement a closed lottery system. If all functionality and requirements are met you will be able to receive 100 points.

Assignment

A group of five friends have decided to set up their own personal lottery. They see this as a fun long term experiment where they might at any point get lucky and win some money. They've tried it for a while with cash and a changing appointed 'lottery boss' but it got messy quickly so now they've decided on a decentralized app.

The front end is taken care of, they just need a solidity developer to build the back end for them. That's you!

Every month, on the first day, each of them picks a number between 1 - 1000 (including 1 and 1000) and sends it with 0.1 Ether to the lottery. The money is added to the pot so if no one wins the amount just keeps growing every month. The way a winner is determined is that the application randomly chooses a number between 1 - 1000 each month. If one of the numbers matches we have a winner! If none of the numbers match the money gets added to the pot and everything starts up again with new numbers the following month. In both cases the participant is notified of the results through an event.

The winner receives the entire pot of money. If multiple people choose the same number *and* that number wins, the money is split evenly between them, they are all winners.

Because they are planning on this being a long term game (the odds of winning are pretty small) they want to keep track of the rounds they've already played. Anyone (inside or outside of the friend group) should be able to check the current pot and the current round.

The contract needs to have an owner and measures need to implemented to change ownership and change the betting amount.

Point distribution

- Correct contract structure and compiles without errors: 5 points
- Rounds are tracked and can be retrieved by anyone: 5 points
- Basic functionality is implemented (no winners: money gets added to pot, 1 winner: winner gets sent money): 30 points
- Functionality for edge case multiple winners is implemented: 15 points
- Owner can change owner and can change betting amount: 10 points
- Random number picker is implemented: 10 points
- Events are implemented correctly: 10 points
- Correct use of visibility modifiers: 5 points
- Inheritance is used: 5 points
- Only participants can participate: 5 points