COMP 4541 – Homework 3

A. K. Goharshady

Release Date: April 16, 2024
Deadline: April 27, 2024 (23:59 HKT)

This homework accounts for 10% of your total grade. You should submit your solutions on Canvas as **one pdf file and two sol files**. Handwritten and scanned solutions will not be accepted since the TAs might be unable to read your handwriting. However, you can draw your figures, if any, by hand. Your solutions must be 100% your own work. All submissions will go through a plagiarism check. The deadline is firm and no extensions will be granted. You will receive feedback on your submission and normally be allowed to resubmit once. Note that the deadline for resubmission might be tight, so try your best to obtain a high mark in your initial submission. We reserve the right to disallow resubmissions if your original submission is of a very low quality or shows a lack of effort.

Important: In each of the exercises, you have to provide both the smart contract (as a sol file) and an explanation of its flow, i.e. who calls which function in which order (in your pdf file). We use Remix to test your contracts. A contract that does not compile will automatically lead to a grade of zero.

Exercise 1

This exercise accounts for 3% of your total grade.

Implement a Solidity smart contract that allows Alice and Bob to play a game of Rock–Paper–Scissors on the Ethereum Blockchain. Each party should first put a deposit of 1 ETH and then 2 ETH is paid to the winner of the game. If the game ties, each party receives 1 ETH back. You can hard-code the addresses of both Alice and Bob at the beginning of your contract, but make sure we can change them easily when testing it. Prove that neither party can cheat in your contract. Also prove that no third-party can tamper with the game as long as both parties have reliable access to the blockchain network.

Submit your code as a single file named `rps.sol`. Make sure your contract is not vulnerable to the attacks discussed in the lectures.

Exercise 2

This exercise accounts for 7% of your total grade.

Consider an open auction in which anyone with access to the blockchain network can bid for a specific item. Assume that the auction runs for 24 hours and anyone can enter a bid in this period. However, the bids must remain secret until the end of the bidding period. After the bidding ends, the winner should be announced and everyone must be able to independently verify that no cheating has taken place.

1. Implement a Solidity smart contract that performs the simple auction mentioned above.
2. What happens if the person with the highest bid refuses to pay? Provide a variant of your contract in which we can ensure payment by the highest bidder. Note that the bids should still remain hidden during the bidding period.
3. How much gas does your contract use? Make sure the amount of gas used by each participant is no more than a fixed constant and *find an upper-bound* for this constant.

Submit your code as a single file named `auction.sol`. Make sure your contract is not vulnerable to the attacks discussed in the lectures.