

University of Edinburgh	<i>Fall 2020-21</i>
Blockchains & Distributed Ledgers	<i>Instructor: Aggelos Kiayias</i> <i>Teaching Assistant: Dimitris Karakostas</i>

Assignment #4 (Total points = 20)

Due: Monday 18.1.2021, 16.30

In this assignment you will create a smart contract that implements a fair swap (see Lecture 8).

First, assume two contracts, C_1 and C_2 , that manage a token. The API of each contract is the same as defined in Assignment #3; therefore, for example, to create C_1 and C_2 you can simply deploy two instances of the contract that you have written for Assignment #3.

Next, assume a user A, who owns at least x tokens on contract C_1 , and user B, who owns at least y tokens on contract C_2 . You should write a new smart contract that implements a special type of fair swap of tokens between A and B. Specifically, your contract should ensure that, during the swap, either both user A receives y tokens on C_2 and user B receives x tokens on C_1 , or neither the balance of A on C_1 nor the balance of B on C_2 are reduced. Note that the contract should perform a swap between exactly x C_1 tokens and y C_2 tokens, regardless of price changes in each token contract before, during, or after the swap.

Your contract should be as fair and secure as possible; any design choices that diverge on either property should be clearly justified. You may assume that the contract implements only one fair swap at a time.

You should submit a PDF report that contains:

- A detailed description of your contract's design.
- A gas and security analysis of your contract.
- A detailed description of how your contract ensures fairness.
- The transaction history of a successful fair swap between two players; you may use either the course's blockchain or Ropsten.
- The code of your contract.