CS480001 / SEC532
Blockchain: Security and Applications
Homework 2
A Secure Token Exchanger Contract

1) Implement your (probably) first ERC20 token and deploy it to the Ropsten test network. The token name must be your “[YourNameSurname]Token” (e.g., ZihniSinirToken). Put 100 tokens to your address.
   a. To see what an ERC20 token, you can read [1].
   b. There are nice solidity examples in [2] including basic implementations of ERC20 tokens. **You can use existing codes as is.**

2) Implement another ERC token with the name “[YourSurnameName]Token” (e.g., SinirZihniToken) and deploy it to Ropsten. Put 200 tokens to your address.

3) Add these tokens to your Metamask wallet. Take a screenshot and paste it to your report. The balances must be visible.

4) Implement a Solidity contract **Exchanger** that registers any number of ERC20 tokens (in your case only the two that you created) and allow an exchange among these two. This contract will have a
   a. **transfer** function which takes two ERC20 token contract addresses (**from_addrs** and **to_addrs** as parameters. In addition, an integer value, **amount**, is also given.
   b. If the balances of the msg.sender hold, the function decreases the **from_addrs** tokens and increases **to_addrs** tokens by **amount**
   c. Deploy it to Ropsten.

5) Use the transfer function to transfer 10 tokens from first to the second ERC20 token you implemented.

6) See the updated balances on Metamask. Take a screenshot and paste it to your report.

7) The report must contain your all the contract codes (in the Appendix) and their addresses (all three of them) in Ropsten.

[1] https://eips.ethereum.org/EIPS/eip-20
[2] https://github.com/OpenZeppelin/openzeppelin-contracts/tree/master/contracts/token/ERC20