**3.1 Undertanding the lecture**

1.a) Describe some advantages of conforming to a token standard like ERC-20 or IStandardToken in a few sentences.

*Conforming to a token standard ensures* ***interoperability*** *between our token and other tokens in the token standard network. This makes buying and trading our tokens easier for end users as well as investors, increasing the appeal of our token.*

*We can also benefit from* ***unified efforts at improving the security*** *of our token standard. The more networks that share the same token standard, the more stakeholders concerned with the token security, the more research is done to improve the token standard security.*

1.b) Describe the purpose of the TimeLockedTransfer contract in a few sentences.

*The TimeLockedTransfer contract allows us to* ***set a time for the contract execution****. This means that even if all conditions are already fulfilled by a certain time, the contract will not, or cannot, be executed until the prescribed time arrives.**It can be used for* ***scheduled transfers*** *or* ***limiting contract accessibility****.*

**3.2 Trying the lecture code**

Download the project ([github.com/Thegaram/cfx-uma-resources/raw/master/cfx-lecture-5.zip 15](http://github.com/Thegaram/cfx-uma-resources/raw/master/cfx-lecture-5.zip)) and import it into Conflux Studio.

In this problem, you will have to deploy three smart contracts on your local development network.

First, deploy a fixed-supply standard coin with the name “CoinA”. Then, deploy a second instance with the name “CoinB”.

Next, deploy an instance of the TimeLockedTransfer contract. Set the unlock time to something greater than 5 minutes.

Send 17 CFX, 22 CoinA, and 12 CoinB to the transfer contract.

After the deadline, withdraw all these tokens from the transfer contract.

Document the process using screenshots and a short description of each step.

**+1. Transfer Factory [OPTIONAL]**

Look at the file TransferFactory.sol. Try to understand what it does and explain it using your own words.

Show us how to deploy and use this contract.

**+1. Cancel transfer [OPTIONAL]**

Modify TimeLockedTransfer.sol in a way that allows the sender to cancel the transfer before the specified deadline.

Provide a way for the sender to re-aquire all their tokens from the transfer contract after cancelling it.