1. Create a script to generate 5 Hedera Testnet accounts (Account1, Account2, Account3, Account4 and Account5). Use these accounts as indicated in the following tasks: 10Marks

· These accounts must not already exist before the start of the test.

· Be sure to note down the accountId and keys of the accounts- you will need these later. You will also need to communicate these account IDs during your response to the certification test.

· Fund the accounts as you see appropriate to cover the costs of your tasks.

2. Create a script that creates a fungible token with the Hedera Token Service belonging to Account1. 15 Marks

· The initial supply should be 350.50 and additional supply can be created by Account2. The maximum supply should 500.

· Create a script to send 25.25 tokens to each of Account3 and Account4.

· Pause the token. Then make another transfer of 1.35 to Account3. The transfer fails. Unpause the transfer and complete the transfer.

3. Download and deploy the solidity bytecode given below using the Hedera Smart Contract Service and Account1. Call function “function1” with parameter “6” and “7” and print the answer you receive. Hint: there are 2 input parameters, and you will receive a return value. Further information is in the ABI file. All parameters are type of uint16. 15 Marks

· Create a second transaction using “function2” and supply the result of “function1” as the input.

· Extra credit: Decode and print the return value from the transactions using ABI decoding.

4. Create a script that creates a scheduled transaction to transfer 10 Hbar from Account1 to Account2. 15 Marks

· Serialize and export 6the transaction to a base64 format and use this as the input to the next step.

· Make a second script that reads in the serialized format and provides the required signature and submit it.

5. Use Account1 as a treasury account so that Account2 can spend 20 Hbar on behalf of Account1. 15 Marks

· Create a transaction that transfers the 20Hbars to Account3.

· Re-run the same operation and show that the allowance has been used and that the second transaction fails. -- SPENDER\_DOES\_NOT\_HAVE\_ALLOWANCE

6. Create a script to create a consensus transaction on the Hedera Consensus Service using Account1. Write the current time in the message of the transaction and submit. 10 Marks

· This time will be used to determine the time of your submission.

Hedera Model Test Questions II

1. Create a script to generate 5 Hedera Testnet accounts (Account1, Account2, Account3, Account4 and Account5).

· Use these accounts as indicated in the following tasks.

· These accounts must not already exist before the start of the test.

· Be sure to note down the account Id and keys of the accounts –you will need these later. You will also need to communicate these account IDs during your response to the certification test.

· Fund the accounts as you see appropriate to cover the costs of your tasks

2. Create a script that generates a fungible token that requires a KYC process to be completed.

· Set a fixed supply of 1000. Associate Account3, but do not KYC it.

· Try to send 12.99 tokens from Account2 to Account3.

· Show that the account is not yet able to participate in the token because it is not been KYC approved.

· Now set the KYC flag on Account3 and retry the transfer

3. Compile the Solidity contract given below using

· Account1 and deploy it to the Hedera Network using the HCS with the ContractCreateFlow method.

· Write a script to call “function1” with the parameters 5 and 6.

· Gather the result and display it in the output.

· Delete the smart contract.

4. Create a script that creates a scheduled transaction of 2 hbar from Account1 to Account2.

· Make a second script that deletes the transaction.

· Print out the schedule information along the way along with the proof that the transfer did not happen.

· Try to execute the transaction and show that it does not work.

5. Create a scheduled transaction with a key list with 3 key (Account1, Account2 and Account3) that requires 2 of the three keys.

· Sign the transaction with Account1. Get the information of the transaction and show that it has not yet been executed.

· Sign the transaction with Account2 and get the information again to show that it has been executed.

6. Create a script to create a consensus transaction on the Hedera Consensus Service using Account1.

· Write the current time in the message of the transaction and submit. This time will b