Implementing the Blockchain Miner

Practical Project

Practical Project: Implementing the Blockchain Miner

This exercise is part of your **"Blockchain Network" practical project**. It will guide you about **implementing the Miner** for our simple educational blockchain network.

Note: you may **mine** on the sample Blockchain Node implementation through its REST API:

  https://stormy-everglades-34766.herokuapp.com.

1. Create a Project Structure for the Miner

Create the initial **project structure** for the **Miner**, setup and configure the **frameworks** and **tools** that you and your team have planned to use. The miner is a simple app, so your code might be just a **single source file**, holding 50-60 lines of code, or it might be more complicated. You are free to decide.

2. Implement the Miner

In an infinite loop repeat the following steps:
  - **Take a mining job** from the Node through its REST API.
  - **Mine the mining job** (change the nonce until you find a hash matching the block difficulty).
  - **Submit the mined job** to the Node in order to build the next block in the chain.

Note that sometimes the submitted mining job will not be accepted by the Node because another miner was faster at finding the block. This is a usual situation and should be expected in the code.

Optionally, you may implement a **multi-threaded miner** (run several threads for each CPU core).

Optionally, you may implement the miner to **take an updated mining job** at certain time intervals, e.g. 1-2 seconds.

3. Experiment with the Miner

Run several blockchain Nodes and several Miners for each Node.

- Learn from the situations that will arise during **mining conflicts**, these may happen when several miners work on the same block.

- Learn about the process of **forking a blockchain network** by running miners in two separated (disconnected) blockchain nodes and connecting those nodes later. The expected behavior should be that **the chain with the most work will win**, and the blocks from the other fork will be lost.

What to Submit?

Submit your **GitHub project URL** on the course website.