

# TP Class Assignment - 25/Mar/2019

---

Each group must answer the questions of the following exercises in the Github area of their group until the end of 12/Apr/2019. For each day of delay, 0.15 points will be deducted from the grade of this assignment.

Note that the virtual machine provided for this course can be used to solve these exercises.

## Exercises

---

### 1. Blockchain

#### Experience 1.1

In this example follow the [Building a blockchain](#) article and the various steps indicated therein.

Please note:

1. To install node.js and npm in the virtual machine provided for this course, execute the following commands in the virtual machine:
  - `sudo su`
  - `curl -sL https://deb.nodesource.com/setup_11.x | bash -`
  - `apt-get install -y nodejs`
  - `exit`
2. You can find the javascript code in the directory [Aula8/koreCoin](#), file `main.experiencia1.1.js`.

#### Question 1.1

In the previous experiment, change the method that creates the Genesis Block, so that the timestamp is today's date and the data included in that block is "Bloco inicial da koreCoin".

#### Question 1.2

In the previous experiment, add a few blocks simulating various transactions in each of them.

#### Experience 1.2

In this example follow the [Let's Build the Tiniest Blockchain in Less Than 50 Lines of Python](#) article and the various steps indicated therein.

Please note:

1. You can find the python code in the directory [Aula8/snakeCoin](#), file `snakecoin.experiencia1.2.py`.

### 2. Proof of Work Consensus Model

## Experience 2.1

This experience is based on Experience 1.1.

In this example follow the [Implementing a simple 'proof of work' algorithm for the Blockchain](#) article and the various steps indicated therein.

Please note:

1. You can find the javascript code in the directory [Aula8/koreCoin](#), file `main.experiencia2.1.js`.

## Question 2.1

In the previous experience, change the difficulty of mining to 2 and see how long it takes, using the Linux time command (or similar on your operating system), for example `time node main.experience2.1.js`. Repeat the example for mining difficulty 3, 4 and 5.

Present the times and conclude on them.

## Experience 2.2

This experience is based on Experience 1.2.

In this example follow the [Let's Make the Tiniest Blockchain Bigger - Part 2: With More Lines of Python](#) article and the various steps indicated therein.

Please note:

1. To install flask in the virtual machine provided for this course, execute the following commands in the virtual machine:
  - `sudo apt-get install python-pip`
  - `pip install flask`
2. You can find the python code in the directory [Aula8/snakeCoin](#), file `snakecoin-server.experiencia2.2.py`.
3. To start the snakeCoin server, execute the following command in the virtual machine:
 

```
python snakecoin-server.experiencia2.2.py &
```
4. To add a transaction (pending transaction) to the snakeCoin blockchain, execute the following command in the virtual machine:
 

```
curl "localhost:5000/txion" \ -H "Content-Type: application/json" \ -d '{"from": "akjflw", "to": "fjlkadj", "amount": 3}'
```
5. To mine the transaction (pending transaction), execute the following command in the virtual machine:
 

```
curl localhost:5000/mine
```

## Question 2.2

1. In the previous experience, what is the 'proof of work' algorithm?
2. Does it seem like an adequate algorithm for mining? Why?

## Engenharia de Segurança course - Project

---

You can use the rest of the class for the Engenharia de Segurança project.