Proyecto - Computación Distribuida

Profesor: Salvador López Mendoza Ayudante: Santiago Arroyo Lozano

13 de noviembre de 2024

Introducción

El proyecto final de la materia Computación Distribuida consiste en desarrollar una blockchain para un sistema de criptomonedas. Este sistema deberá ser capaz de manejar múltiples procesos en Elixir que representen a los usuarios, quienes podrán enviarse mensajes, alcanzar un consenso y detectar y eliminar procesos maliciosos que intenten alterar la blockchain.

Evaluación

La calificación del proyecto se dividirá en dos rubros:

- Parte práctica, esto equivale a la calificación total del proyecto (10 % de la calificación final del curso)
- Parte teórica, equivalente a su tercer examen de la materia. Es decir, esta parte del proyecto será ponderada como su examen de consenso. (12.5 % de su calificación final)

En total **el proyecto vale 22.5 %** de su calificación final. Cualquier código que se entregue y que no se pueda compilar tiene cero en automático. Se evaluarán las mismas cualidades que en las prácticas, limpieza de código, documentación y funcionalidad.

Desarrollo

El proyecto deberá ser implementado en Elixir y no se limita a los alumnos a solamente implementar las funciones y módulos descritas en este documento, cualquier propuesta es válida y pueden implementar cuantas funciones auxiliares necesiten.

No es necesario implementar ningún tipo de persistencia o validación de la cartera. Una vez que el programa termina, deja de existir la blockchain, no guardaremos en ningún archivo o base de datos el estado. Por la parte de la cartera, no revisaremos si las transacciones son válidas preguntando si hay saldo suficiente ni nada de eso. Lo que queremos evaluar es que exista un consenso de la blockchain y que todos los nodos confirmen y registren las mismas transacciones a realizar.

Se espera poder identificar servicios o módulos dentro de su código que cumplan con las siguientes funciones:

1. Crypto - Encargado de hacer los hasheos. Este módulo les será proporcionado para que lo utilicen
2. Block - Estructura representando un bloque dentro del la blockchain
3. BlockChain - La blockchain completa y un validador de cada bloque a insertar, tolerante a fallas.
4. Main - Un código manejador de varios procesos en Elixir que permita transacciones entre nodos.

En resúmen, el main spawneará varios procesos y se simularán transacciones benignas y malignas en la blockchain, la cual debe validarse a si misma y cada bloque nuevo que se quiera inserta. Cada bloque a su vez debe encargarse de generarse de la manera correcta a partir de un conjunto nuevo de datos que se quiera insertar en la blockchain. Crypto solo será un modulo de utilidades.

Bloque

Un bloque es la estructura que va a contener los datos de las transacciones que se realizan, almacenadas como una cadena de texto o estructura de datos. Debe poder mostrarse en pantalla en la terminal y debe contener los siguientes atributos:

- `data` - El contenido del bloque como tal. Puede ser un mensaje, una estructura de datos, etc
- `timestap` - La hora y fecha en el que fuera generado el bloque
- `prev_hash` - El hasheo del bloque anterior
- `hash` - El hasheo de este bloque

Además el bloque debe cumplir las siguientes funciones:

- `new/2`: Dados datos y un hash anterior creamos un nuevo bloque
- `valid?/1`: Validamos un bloque dado
- `valid?/2`: Validamos si un par de bloques secuenciales son validos

Blockchain

Podemos ver al blockchain como una lista de bloques. Tiene la responsabilidad de estarse validando a si misma constantemente y validar cada bloque que queramos insertar a esta. Recordemos que no se pueden modificar ni eliminar bloques del blockhain, solo añadir cosas nuevas o desechar completamente una blockchain comprometida. Las funciones que debe cumplir son

- `insert/2`: Insertamos nuevos datos a la blockhain dada.
- `valid?/1`: Validamos la blockchain dada.

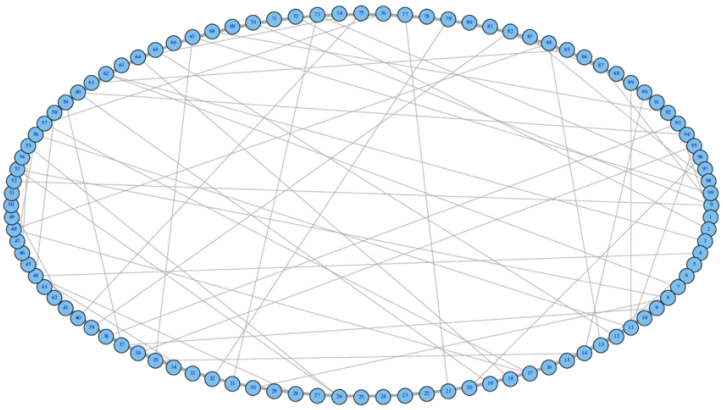
Main

El proyecto debe correr desde un solo módulo manejador, específicamente desde una sola función **main** que reciba los siguientes parámetros:

1. `n`: Número de nodos en la red
2. `f`: Número de procesos bizantinos (procesos que simplemente van a tratar de alterar la blockchain)

Pueden asumir que $n > 3f$.

La forma en que la red asigna vecinos debe basarse en el modelo watts y strogatz:



La red debe tener un coeficiente de agrupamiento mayor a 0.4 antes de comenzar el proceso.

Se espera que el evaluador pueda hacer algo de la forma:

```
iex> blockchain = Main.run(10, 1)
[
    %{Block1 ...}
    %{Block2 ...}
    %{Block3 ...}
    ....
]
```

Se espera que dicha función:

- Inicialice los procesos
- Construya la red siguiendo el modelo y los parámetros indicados
- El evaluador podrá interactuar con la blockchain desde el código o la terminal para insertar bloques nuevos y verificar que sí se esté propagando

Bizantinos

Los procesadores bizantinos simplemente deben generar bloques basura. El evaluador revisará que estos bloques no sean incluidos en la blockchain.

Entrega

La entrega se va a dividir en dos etapas, cada una tendrá su apartado particular en el Classroom para entregarse debidamente.

Primer Entrega

22 de Noviembre 2024

Deberán entregar el código en Elixir que modele su block chain, todavía no se espera el funcionamiento completo de toda la red ni resistencia a fallas. Se espera que ya tengan una implementación del manejador y el modelo de Waltz.

Lo más importante de esta entrega, que se calificará como su examen 3 es una propuesta escrita de como planean implementar el consenso en esta red. El algoritmo de consenso es de libre elección y pueden basarse en algo ya existente, pero para corroborar que no se compliquen esto es lo primero que esperamos que entreguen para poder afinar los detalles antes de la entrega final.

Cualquier error que se detecte en esta parte se les hará saber y podrá ser corregido para la segunda entrega.

Segunda Entrega

29 de Noviembre 2024

Se calificará el proyecto en su totalidad. No se permitirán correcciones ni entregas después de esta fecha.

El proyecto debe llevar un **Readme** que contenga los nombres de los integrantes del equipo (Máximo 3 personas) así como el procedimiento para ejecutar la práctica y cualquier otro comentario importante respecto a su funcionamiento.
Mucho éxito!