Universidad Nacional Autónoma de México
Licenciatura en Ciencias de la Computación, Facultad de Ciencias
Computación Distribuida
Examen 3
Información General

Profesor:	Miguel Angel Piña Avelino
Ayudantes:	Diego Estrada Mejía, Daniela Susana Vega Monroy
Laboratoristas:	Luis Fernando Fong Baeza, Pablo Gerardo González López
Fecha de Entreg:a	30 de enero del 2022
Alumno:	Montiel Manriquez Ricardo

Problemas

1. En el contexto de blockchains, describe con detalle la ejecución del algoritmo de prueba de trabajo (algoritmo de consenso) de Bitcoin.

   En Blockchain un algoritmo de consenso es el mecanismo usado, por una red Blockchain, para seleccionar el estado correcto de un registro después de realizar una transacción. De esta manera lo que indique el algoritmo de consenso se convierte en la verdad que todos los nodos deben seguir.

   Entonces para su ejecución tenemos que:

   - Se calcula la dificultad objetivo d, dicha d se calcula cada 2016 bloques de modo que el tiempo promedio para agregar un bloque sea de 10 minutos. En este paso se asigna la tarea computacional mente costosa, dicha tarea debe cumplirse para recibir el premio.
   - Cada minero en la red, propondrá su bloque X de transacciones, en este paso se intentara resolver el problema que consta de acertar un numero hash, dicho resultado se obtiene de una función $h(x)$ que calcula el hash y tiene que cumplir que si se hace un cambio en el contenido del bloque, se pueda generar un valor totalmente distinto al anterior. Este paso se repetirá hasta que alguna propuesta cumpla con la condición: $h(x) \leq d$
   - Cuando no se cumpla la condición, el minero deberá modificar el nonce del bloque el cual es el único valor que puede ser modificado después de que se creo el bloque y antes de ser añadido a la cadena de bloques.
   - Si la condición se cumple, el minero enviara sus resultados (bloque) a la red, para entrar a proceso de verificación, con los otros nodos. En este punto cada nodo se encarga de verificar que el bloque propuesto es exitoso, es decir si satisface la dificultad del objetivo. Si lo hace, el bloque se a˜nade a la lista ligada de bloques
   - Finalmente cuando se confirma que se ha añadido un bloque con la transacción, hay que esperar a que se a˜nadan algunos bloques adicionales a la blockchain. Es posible que exista un fork de la blockchain a causa de que dos mineros resolvieron el problema con distinto bloque pero al mismo tiempo. Como existen los retardos en la propagación los participantes pueden tener una historia distinta entre si pero la solución es que tomaremos la cadena con sufijo mas larga, a partir de donde se genero el fork.

2. Modifica la descripción del algoritmo anterior para permitir la ejecución de contratos inteligentes.

   Se tendría que implementar sobre la infraestructura del Blockchain, que comienza cuando existe una transacción y sus resultados son almacenados en el block chain.

   En este punto, cada nodo se encarga de verificar la ejecución del contrato, es decir, un programa con las especificaciones o instrucciones que tienen que ser las mismas a las del bloque propuesto por un minero. La información queda almacenada en cada nodo de la Blockchain, en donde se duplica la información teniendo la seguridad que esta no pueda ser alterada por nadie. Esto porque cada vez que un bloque es confirmado y este se añade a la lista ligada, se comunica con todos los nodos y se añade a la copia que cada nodo almacena.

3. Describe con tus palabras el problema del acuerdo del k-conjunto y el problema del consenso simultaneo.

   - Acuerdo del k-conjunto:

     Es una abstracción de muchos problemas de coordinación en un sistema distribuido que puede sufrir fallas en el proceso. Cada proceso comienza con un valor de entrada y debe decidir irrevocablemente sobre un valor de salida, de modo que los procesos correctos decidan un total de, como máximo, k valores. El conjunto de valores de decisión permitidos se especifica mediante una condición de validez que restringe las decisiones de los procesos correctos en función de los valores de entrada y si se producen fallas durante la ejecución del protocolo.
   - Consenso simultaneo:

     Este problema basicamente surge porque un proceso falla, dado que falla significa que es estable y este queda inactivo, pero un estado que no ha fallado es inestable, ya que podria fallar en cualquier momento. Por esta razon es que el peor de los casos se da cuando ninguno de sus nodos falla, ya que todos son inestables o la ocurrencia de unos pocos procesos fallidos hacen que decidir en una ronda temprana sea muy difícil de obtener. En pocas palabras es un algoritmo que requiere que los procesos decidan en la misma ronda, y esta tiene que ser lo mas temprana posible, sin importar el numero de procesos fallidos.

4. En que consiste el problema de la coloración de vértices. Menciona a grandes rasgos en que consiste el algoritmo de $\triangle + 1$ coloración en gráficas arbitrarias.

   El problema de coloración de vértices es el problema de asociar una coloración válida con una gráfica dada. es decir que la asignación de un color c(v) a cada vértice tal que cualesquiera 2 vértices adyacentes tengan diferente color.

   $\triangle + 1$ Es un algoritmo para colorear una grafica arbitraria de grado maximo $\triangle$ con $\triangle + 1$ colores y con un tiempo de $O(\triangle logn)$

   El algoritmo esta basado en un procedimiento recursivo $RecursiveColor(X)$, en donde X es una cadena binaria y $U(X)c = V$ denota la coleccion de vertices con un Id que tiene subfijo X. Se aplica $RecursiveColor(X)$ a $U(X)$ y se obtine la coloracion para los vertices $U(X)$ con $\triangle + 1$ colores.

5. Considera la exposición de los relojes físicos y explica en que consiste el problema del shift. ¿Qué soluciones se han propuesto para resolver este problema?

El problema se encuentra en la sincronización de relojes, ya que requerimos que los procesadores tengan los relojes muy cercanos entre ellos para su comunicación porque el hardware de los relojes no es controlado por los procesadores. Por lo que se implemento un componente que pudiese modificarlos 'adj(i)' y durante la sincronización se puede ajustar el reloj con este valor. Por lo que el reloj que fue ajustado lo tendremos definió como el reloj actual + el reloj de hardware.

6. Describe a grandes rasgos el algoritmo **EIG** para consenso bizantino.

   A grandes rasgos podemos definir que el algoritmo en arbol en tres pasos.

   - Los procesadores envían y retransmiten valores para el número de rondas.
   - Registrar los valores que reciben a lo largo de varias rutas de comunicación en una estructura de datos llamada árbol EIG.
   - Al final, utilizan una regla de decisión comúnmente acordada y deciden en función de los valores registrados en sus árboles.

   Explicando el algoritmo tendriamos que:

   La primera parte del algoritmo consiste en la inicialización del árbol, seguido de rondas sincrónicas en las cuales pi calcula los valores de su representación local del árbol. Por lo tanto, esta parte es la recopilación de información y en cada ronda cada proceso sigue los siguientes pasos: Enviar un mensaje (fase) y recibirlo

   La segunta parte del algoritmo es local y no implica comunicacion alguna, donde se hace el calculo del valor decidido y se procede desde sus hojas hasta su raiz.

7. Explica el teorema **CAP** y menciona las soluciones existentes en el modelo asíncrono.

   El teorema CAP, también llamado Conjetura de Brewer, enuncia que un sistema distribuido puede entregar solo dos de tres características deseadas: CONSISTENCIA, DISPONIBILIDAD y TOLERANCIA A LA PARTICIÓN, (CAP, en ingles).

   - Consistencia:
     Significa que todos los clientes ven los mismos datos al mismo tiempo, independientemente del nodo al que se conecten. Para que esto suceda, siempre que se escriban datos en un nodo, se debe reenviar o replicar al instante a todos los demás nodos del sistema antes de que la escritura se considere 'satisfactoria'.
   - Disponibilidad:
     Significa que cualquier cliente que realiza una solicitud de datos obtiene una respuesta, incluso si uno o más nodos están inactivos. Otra forma de indicar esto: todos los nodos activos del sistema distribuido devuelven una respuesta válida para cualquier solicitud, sin excepción.
   - Tolerancia de partición:
     Una partición es un quiebre de las comunicaciones dentro de un sistema distribuido: una conexión perdida o temporalmente retardada entre dos nodos. La tolerancia a las particiones significa que el clúster debe continuar trabajando a pesar de las interrupciones de comunicación que se produzcan entre los nodos del sistema.

Para las soluciones en el Modelo Asincrono es imposible proporcionar estas tres propiedades, ATOMICIDAD, DISPONIBILIDAD y TOLERACIA DE PARTICION, pero si se puede lograr dos de estas tres propiedades.

- Atomico y Tolerante a Particiones.
  Si no se requiere disponibilidad, entonces es fácil lograr datos atómicos y tolerancia de partición. El sistema trivial que ignora todas las solicitudes es un ejemplo.
- Atomico y Disponible.
  Si no hay particiones, es claramente posible proporcionar datos atómicos disponibles. Los sistemas que se ejecutan en intranets y LAN son un ejemplo de este tipo de algoritmos.
- Disponible y Tolerante a Particiones.
  Es posible proporcionar alta disponibilidad y tolerancia a la partición, si no se requiere consistencia atómica. Los cachés web son un ejemplo de una red débilmente consistente.

8. Explica como funcionan los esquemas de shortest path routing y full tables routing.
9. Siguiendo la idea del "Conocimiento común" descrita en la presentación de *Knowledge in distributed systems*, ¿Por qué no se puede solucionar el problema del ataque coordinado?
10. Explica el ejemplo del ataque coordinado usando topología que fue mostrado en la presentación de *Topología y sistemas distribuidos.*