

Hashes y Hashes Criptograficos

Prof. Ernesto Rodriguez

Universidad del Itsmo

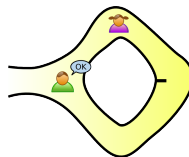
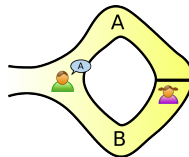
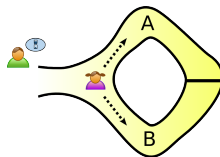
erodriguez@unis.edu.gt

Zero-Knowledge Proof

- Permiten demostrar el conocimiento de un secreto sin tener que revelar el secreto.
- Ejemplos:
 - Demostrar saber una contraseña sin tener que revelar la contraseña.
 - Demostrar el conocimiento de un ciclo Hamiltoniano
 - Demostrar tener fondos suficientes para una transacción sin revelar la cantidad de fondos ([Monero](#))

La Cueva de Ali-Baba

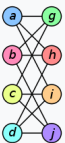
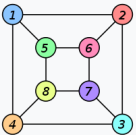
- 1 Una cueva con una puerta en medio que se abre con una palabra magica.
- 2 Peggy conoce la palabra y se lo quiere demostrar a Victor sin revelar la palabra
- 3 Peggy ingresa a la cueva y toma el camino A o B al azar. Victor no sabe que camino tomo.
- 4 Victor le dice a Peggy, al azar, por que camino debe regresar.
- 5 Peggy regresa por el camino indicado, utilizando la palabra mágica si es necesario.

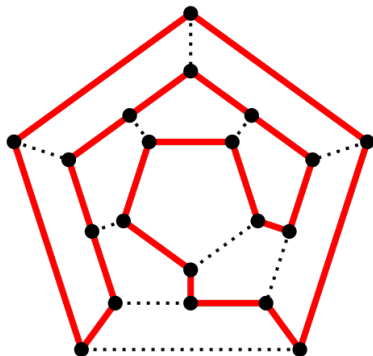


Camino Hamiltoniano

- 1 Peggy conoce un camino ñ Hamiltoniano de un grafo \mathcal{G}
- 2 Peggy se lo quiere demostrar a Victor sin revelar el camino.
- 3 Peggy crea un grafo \mathcal{H} , que es isomorfico a \mathcal{G} , en otras palabras, para cada vertice $g \in \mathcal{G}$ se define una biyeccion a los vertices $h \in \mathcal{H}$.
Peggy no revela este isomorfismo.
- 4 Peggy se compromete a dicho isomorfismo, posiblemente escribiendolo en un papel de tal manera que ella no lo puede cambiar.
- 5 Victor debe escoger entre ver el isomorfismo o ver el camino Hamiltoniano:
 - Si Victor desea ver el isomorfismo, Peggy lo revela
 - Si Victor desea ver el camino Hamiltoniano, Peggy revela dicho camino en el grafo \mathcal{H} y solamente las esquinas utilizadas en el camino Hamiltoniano.

Camino Hamiltoniano

Graph G	Graph H	An isomorphism between G and H
		$f(a) = 1$ $f(b) = 6$ $f(c) = 8$ $f(d) = 3$ $f(g) = 5$ $f(h) = 2$ $f(i) = 4$ $f(j) = 7$



- Son un mapeo de objetos de un conjunto \mathcal{U} a un conjunto más pequeño $\mathcal{M}_k := \{0, 1, \dots, k\}$
- El conjunto \mathcal{U} puede ser cualquier colección de objetos: strings, matrices, personas, ect.
- Debido a que el conjunto \mathcal{U} es más grande que el conjunto \mathcal{M}_k , necesariamente existen *colisiones*
- Una *buena función de hash* distribuye sus colisiones de forma uniforme.

Aplicaciones de Hashes

- Diccionarios
- Almacenamiento seguro de contraseñas
- Validación de integridad
- Proof of work

Hashes criptograficos y no-criptograficos

Un hash criptografico es dificil de invertir!

- Los hashes no criptograficos solo evitan colisiones
- Dado un hash criptografico y un valor $m \in \mathcal{M}_k$, es dificil obtener el valor original $u \in \mathcal{U}$ que produjo dicho m . Idealmente, es tan dificil como una busqueda lineal.
- Un hash criptografico se debe comportar como una *asignación aleatoria* en lo más posible.
- Los hashes no-criptograficos son más eficientes.

Ejemplos de Hashes criptograficos

- MD5: uno de los primeros, ahora no se considera seguro
- SHA-256: El hash criptografico más usado. Se utiliza en el Blockchain de Bitcoin
- Scrybd: Intenta ser el más caro de invertir. Aprovechando el costo de memoria ram. Se utiliza con los Litecoins.

Nota: La distinción entre hash criptografico y no criptografico depende de la *intención* de dicho hash. No su rendimiento.