RESOLUCIÓN PRUEBAS CRIPTO

FUNCIONAMIENTO DEL ALGORITMO DE EUCLIDES.

Alg. de Euclides es un método antiguo y eficiente para calcular el máximo común divisor (MCD).

Pasos:

Se tiene 2 números enteros positivos a y b.

Se divide el número mayor entre el menor (por ejemplo [a/b]), del cual se obtiene el cociente c, y un resto r. Si el resto r=0, entonces el mcd(a,b)=b. Si no es cero, dividir el divisor, c, entre el resto r, obteniendo otro conciente, cc, y otro resto rr. Si rr=0, entonces mcd(a,b)=r. Si no es 0 nuevamente se divide divisor entre resto. Así sucesivamente. Así el mcd(a,b) es el último resto distinto de 0 que se obtiene.

Ejemplo: mcd(721,448)

721 = 448*1 + 273 ///// 448= = 273*1 +175 ///// 273= 175*1 +98 ///// 175 = 98*1 + 77///// 98=77*1/////
77=21*3+14 ///// 21=14*1+7/////14=7*2+0

Entonces mcd(721,448)=7

QUE ES UNA FUNCIÓN DE EULER Y UN EJEMPLO

La función de Euler (φ): Dado $n \varepsilon N$, se define $\varphi(n)$ como la cantidad de números naturales menores o iguales que n que son primos relativos con el propio n

Representación:

$$\varphi(n) = \left|\{m \varepsilon Z^{+};\ m < n \wedge mcd(m,n) = 1\}\right|$$

Importante: (Dos números enteros a, b son primos relativos si mcd(a,b)=1)

Ejemplo:

Calcular $\varphi(90)$

$90 = 3^2 \cdot 2 \cdot 5$

Si $mcd(m,n) = 1, \varphi(n \cdot m) = \varphi(n)\varphi(m)$

- $\varphi(90) = \varphi(3^2 \cdot 2 \cdot 5) = \varphi(3^2)\varphi(2)\varphi(5)$

Si p es primo: $\varphi(p) = p - 1;\ \varphi(p^a) = p^{a-1}(p-1)$

- $\varphi(3^2) = 3 \cdot 2; \varphi(2) = 1; \varphi(5) = 4$

$$90 = 3^2 \cdot 2 \cdot 5$$
$$\varphi(90) = \varphi(3^2 \cdot 2 \cdot 5) = \varphi(3^2)\varphi(2)\varphi(5)$$
$$= 6 \cdot 1 \cdot 4 = 24$$

TEOREMA DE FERMAT

Si p es un número primo, entonces, para cada número natural a, con a>0 , coprimo con p , $a^{p-1} \equiv 1 \pmod p$

ALGORITMO RSA

Los pasos son:

- Generar dos números primos muy grandes, p y q
- Hacer n=p*q
- Calcular z=(p-1)*(q-1)
- Elegir un pequeño número k, co-primo de z, de modo que el Máximo Común Divisor entre z y k sea 1, con 1<k<z
- Encontrar un número j tal que el (j mod z) sea 1
- Publicar k y n como la clave pública.
- Guardar j como la clave privada.

Luego, para el cifrado se usan estas expresiones, basadas en los anteriores valores:

$mensaje\_cifrado = (mensaje\_plano)^k \ mod\ n$

$mensaje\_plano = (mensaje\_cifrado)^j \ mod\ n$

Ejemplo:

Mensaje en texto plano (sin cifrar) denotado por P, será cifrado utilizando esta ecuación matemática:

$P^k = E(mod\ n)$

Donde: **"P"** es el mensaje en texto plano, **"n"** y **"k"** son la clave pública, **"E"** es el mensaje cifrado

Sustituyendo los valores:

14^7 = E (mod 33) Donde 14^7 = 105413504

Resultado divido para 33 (n) => 105413504 / 33 = 3194348.606

Calcular el resto entero

3194348.606 - 3194348 = 0.606 0.606 * 33 = 19.998 ~ 20

Por lo tanto, **E=20**, que es el texto cifrado

DESCIFRADO DEL MENSAJE

El destinatario debe tener su clave privada j, que en nuestro ejemplo, vale j=3.

Realizar esta operación matemática:

E^j = P (mod n)

Donde: **"E"** es el mensaje cifrado, **"j"** la clave privada, **"P"** el mensaje en texto plano, **"n"** es parte de la clave pública del destinatario.

20^3 = P (mod 33)

20^3 = 8000 Divido para 33 (n) => 8000/33 = 242,242242…

El resultado entero del cociente es: 242*33 = 7986

Por lo que el resto, P, o el mensaje original en texto plano, será: 8000-7986 = **14**

CRIBA DE ERATÓSTENES Y SUS PROPIEDADES

Algoritmo que permite hallar todos los números primos menores que un número natural dado n. Se forma una tabla con todos los números naturales comprendidos entre 2 y n, y se van tachando los números que no son primos

Ejemplo: números primos menores de 20.

1: listar los números naturales comprendidos entre 2 hasta n (por ejemplo 20)

2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20

2: Se toma el primer número no rayado, como número primo.

2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20

3: Se tachan todos los múltiplos del número que se acaba de indicar como primo.

2,3,5,7,9,11,13,15,17,19

4: Si el cuadrado del 1er número que no ha sido rayado ni marcado es inferior a 20, entonces se repite el segundo paso. Si no, el algoritmo termina, y todos los enteros no tachados son declarados primos.

Como $3^2 = 9 < 20$, se vuelve al segundo paso: 2,3,5,7,11,13,17,19

En el **4to paso**, el 1er número que no ha sido tachado ni marcado es 5. Como su cuadrado es mayor que 20, el algoritmo termina y se consideran primos todos los números que no han sido tachados.

Como resultado se obtienen los números primos entre 2-20: **2, 3, 5, 7, 11, 13, 17, 19.**

SÍMBOLO DE LEGENDRE

Símbolo de Legendre es usado como una forma de clasificar los enteros "**a"** respecto a un primo impar **"p"** bajo el criterio de residuos cuadráticos. Respecto al módulo **p**, un entero **a** es ya sea múltiplo de **p** o bien es primo con **p**. Y si es primo con **p** entonces es residuo cuadrático de **p** o bien no lo es (en cuyo caso se dice que es un residuo no cuadrático). El símbolo es el siguiente: (a/p)

Y es : 0 si a es múltiplo de **"p"**, 1 si a es residuo cuadrático de **"p",** -1 si a es residuo no cuadrático de **"p"**

PROPIEDADES

El símbolo de Legendre satisface algunas propiedades:

- $\left(\frac{q}{p}\right)\left(\frac{p}{q}\right) = (-1)^{\frac{p-1}{2}\frac{q-1}{2}}$ para todo par de primos impares p y q
- $\left(\frac{ab}{p}\right) = \left(\frac{a}{p}\right)\left(\frac{b}{p}\right)$

Ejemplo:

$$x = 2 \Rightarrow 2^2 \equiv 1 \pmod 3 \Rightarrow \left(\frac{1}{3}\right) = 1.$$

CRIBA CUADRÁTICA

Es un algoritmo de factorización de enteros, el segundo método más rápido conocido (después de la criba general del cuerpo de números). Es el más rápido para enteros que tienen 100 o menos dígitos decimales, y es considerado mucho más sencillo que la criba de cuerpos numéricos. Es un algoritmo de factorización de propósito general, lo que significa que su tiempo de ejecución únicamente depende el tamaño del entero a ser factorizado, y no sobre una estructura especial o propiedades.

Ejemplo:

Factorizar n=87453
Sea B=30 el máximo rango de números de la base
Sea (–C,C)= (-35,35) el rango de números aleatorios
Paso 1: Encontramos los números que son residuo cuadrático de n.

P	2	3	5	7	11	13	17	19	21	23	29
Jacobi(p,n)	1	1	-1	-1	-1	1	1	1	-1	-1	1

Paso 2: m=√n, m= 295
Paso 3 ,4 ,5,6. Donde obtendremos esta matriz de números lisos

x	f(x)	-1	2	3	13	17	19	29
265	-17238	1	1	1	0	1	0	0
278	-10179	1	0	1	1	0	0	1
296	153	0	0	0	0	1	0	0
299	1938	0	1	1	0	1	1	0
307	6786	0	1	0	1	0	0	1
316	12393	0	0	0	0	1	0	0

Paso 7: Una posible solución que cumple que su suma sea cero es fila 3 y fila 6.

x	f(x)	-1	2	3	13	17	19	29
296	153	0	0	0	0	1	0	0
316	12393	0	0	0	0	1	0	0

Paso 8: X=296*316 = 6073
Paso 9: Y^2 = 153*12393, Y= 1377
Paso 10: mcd(X-Y,n)= 587 y mcd(X+Y,n)= 149
Para verificación 587*149= 87463

RAÍZ DE DOS

Por fracciones continuas

Sea raíz cuadrada de dos: r=$\sqrt{2}$, su parte entera vale 1, así que $a_0 = 1$ y $x_0 = 1/(\sqrt{2}-1)$. Ahora bien, utilizando la identidad $(\sqrt{2}-1)(\sqrt{2}+1) = 1$, tenemos que $x_0 = \sqrt{2}+1$. Por tanto $a_1 = 2$ y $x_1 = \sqrt{2}+1$. Concluimos que todos los $a_k = 2$ a partir de k=1 valen 2 y todos los x_k valen $\sqrt{2}+1$. El desarrollo en fracción continua es, por tanto:

$\sqrt{2} = [1; 2, 2, 2, \ldots]$.

El algoritmo lee n, a=1, b=1, entra en un bucle while (b<n) haciendo b=a+b; a=2b-a al final solo imprime a/b

CRIBA DE FRACCIONES CONTINUAS

Un algoritmo de factorización prima que usa residuos producidos en la fracción continua de $\sqrt{mN}$ algunos elegidos adecuadamente m para obtener un número cuadrado . El algoritmo resuelve $x^2 \equiv y^2 (mod\ n)$ encontrando un m para el cual $m^2 (mod\ n)$ tiene el límite superior más pequeño. El método requiere (por conjetura) sobre los $exp\ (\sqrt{2\ ln\ n\ ln\ ln\ n}$ pasos, y fue el algoritmo de factorización de primos más rápido en uso antes de que se desarrollara la criba cuadrática, que elimina los 2 de la raíz cuadrada

CURVAS ELÍPTICAS

El método de autorización de curva elíptica, ECM o tb llamado método de curva elíptica de Lenstra, es un algoritmo de factorización que calcula un gran múltiplo de un punto en un módulo de curva elíptica aleatoria, del número a factorizar **"N"**. Se tiende a ser más rápido que la factorización Pollard *rho* y Pollard *p* -1.
Es un rápido algoritmo de tiempo de ejecución sub-exponencial para la factorización de enteros que emplea curvas elípticas. ECM es considerado un algoritmo de factorización de propósito especial así como el más adecuado para encontrar factores pequeños. Es el mejor algoritmo para divisores que no superen los 20 a 25 dígitos decimales, así como su tiempo de ejecución está dominado por el tamaño del factor más pequeño "**p"** en lugar de por el tamaño del número **"n"** a ser factorizado. ECM se usa para eliminar factores pequeños de un entero muy grande con muchos factores; si el entero resultante todavía es compuesto, entonces solo tiene factores grandes y es factorizado mediante el uso de técnicas de propósito general. Incrementando el número de curvas probadas se mejoran las posibilidades de encontrar un factor, pero no son lineales con el incremento en el número de dígitos.
Explique porque si n=axb es recomendable cribar alrededor del valor de la raíz cuadrada de n

Es más probable que en el intervalo cercano a la raíz cuadrado de n existan los valores compuestos de n, reduce el intervalo, los pasos y el tiempo empleado por algoritmos de factorización de números

EXPLIQUE LA DIFERENCIA ENTRE NÚMEROS PERFECTOS Y NÚMEROS DE MERSENNE

Un **Número de Mersenne** es un número entero positivo m que es una unidad menor que una potencia entera positiva de 2: $M_n = 2^n - 1$. Un número de mersenne no siempre es primo, se cumple con la condición de primalidad si n también es primo

Un **número primo de Mersenne** es un número de Mersenne que es primo. Se cumple que todos los números de Mersenne, $M_n = 2^n - 1$ que sean primos también tendrán n prima (aunque no toda n prima vale; no es una condición suficiente que n sea prima para que M_n lo sea).

Un **número perfecto** es un número natural que es igual a la suma de sus divisores propios positivos. Dicho de otra forma, un número perfecto es aquel que es amigo de sí mismo. Ejm: 6 es un número perfecto porque sus divisores propios son 1, 2 y 3; y 6 = 1 + 2 + 3.

EXPLIQUE LA DIFERENCIA ENTRE NÚMEROS PSEUDOPRIMOS Y PSEUDOPRIMOS FUERTES

Los **pseudoprimos fuerte** son aquellos números que no siendo primos, verifican el test de base dos: Siendo "a" perteneciente a los números enteros "a" es pseudoprimo si se verifica que: $2^a = 2 \pmod a$ (2 elevado a"a" es congruente a 2 módulo a)

Los **pseudoprimos** son aquellos números que no siendo primos, verifican el test de base *b*: Siendo *n* perteneciente a los números enteros, se dice que *n* es pseudoprimo respecto la base *b* si es compuesto y además verifica la congruencia: $b^{n-1} \equiv 1 \ (mod\ n)$

ENUNCIAR LA FÓRMULA DE LA OBSERVACIÓN DE FERMAT

Si p es un número primo, entonces, para cada número natural a, con a>0 , coprimo con p , $a^{p-1} \equiv 1 \pmod p$

SEA ϕ LA FUNCIÓN DE EULER EXPLIQUE PORQUE $\phi(p) = p - 1$ SI P ES UN NÚMERO ENTERO POSITIVO PRIMO

Es una de las propiedades de la función de euler para números enteros positivos primos, es decir siempre se cumplira si tiene todas las condiciones previas

ENUNCIE EL TEOREMA CHINO DE RESIDUO

Sean $n_1, n_2, \ldots, n_k$ enteros positivos tales que $mcd(n_i, n_j) = 1$ para $i \neq j$ y sean $a_1, a_2, \ldots, a_k$ enteros arbitrarios. Entonces el sistema $x \equiv n_1 a_1 \quad x \equiv n_2 a_2 \quad \ldots \quad x \equiv n_k a_k$ tiene al menos una solución. Si el sistema es soluble, cualesquiera dos soluciones del sistema son congruentes módulo $\Pi_{i=1}^{k} n_i$.

EXPLIQUE QUÉ SE ENTIENDE POR POLLARD RHO

Es un algoritmo especializado de factorización de números enteros.Es efectivo factorizando números compuestos que tengan factores pequeños.

Conceptos utilizados en el algoritmo Rho de Pollard:

Se dice que dos números x e y son congruentes módulo n ($x \equiv y \ mod\ n$) si su diferencia absoluta es un múltiplo entero de n, o cada uno de ellos deja el mismo resto cuando se divide por n.

El divisor común más grande es el número más grande que se divide uniformemente en cada uno de los números originales.

Paradoja de cumpleaños: la probabilidad de que dos personas tengan el mismo cumpleaños es inesperadamente alta, incluso para un grupo pequeño de personas.

El algoritmo de búsqueda de ciclo de Floyd: si la tortuga y la liebre comienzan en el mismo punto y se mueven en un ciclo tal que la velocidad de la liebre es el doble de la velocidad de la tortuga, entonces deben cumplir en algún punto.

EXPLIQUE QUÉ ENTIENDE POR RECIPROCIDAD CUADRÁTICA

Relaciona la solubilidad de dos congruencias de segundo grado relacionadas, donde p y q son números primos impares $x^2 \equiv p(mod\ q)$ $y^2 \equiv q(mod\ p)$

Si ninguno de los primos p o q pertenece a la sucesión 4k+1 entonces una de las congruencias tiene solución si y sólo si la otra no tiene solución. Si alguno de los primos pertenece a la sucesión 4k+1 entonces o bien ambas congruencias tienen solución o bien ninguna de las dos tiene solución.

EXPLIQUE LA UTILIDAD DE LAS CRIBAS, CITE DOS CRIBAS QUE CONOCE

Una criba es un algoritmo que permite descomponer números compuestos grandes en sus factores primos. Criba de Eratóstenes y Cuadrática

ALGORITMO DE BHASKARA-BROUNCKER

Es un algoritmo que nos permite aproximar la raíz cuadrado de un número, se enuncia de la siguiente manera:

Sea Q un entero no cuadrado. El algoritmo Bhaskara-Brouncker da aproximaciones sucesivas de √Q como a i / b i , donde a 0 = b 0 = 1, y

$a_{i+1} = a_i + b_i Q \; ; \; b_{i+1} = a_i + b_i$

Ahora, al configurar $x_i = a_i/b_i$, obtenemos la siguiente fórmula sorprendente para aproximaciones sucesivas a √Q: $x_{i+1} = (x_i + Q)/(x_i + 1)$ con $x_0 = 1$.

Ejemplo. Encuentre √5 usando el algoritmo Bhaskara-Brouncker:

5/2 → (5 + 2 (5)) / (5 + 2) = 15/7

15/7 → (15 + 7 (5)) / (15 + 7) = 50/22

50/22 → (50 + 22 (5)) / (50 + 22) = 160/72

160/72 → (160 + 72 (5)) / (160 + 72) = 520/232

520/232 → (520 + 232 (5)) / (520 + 232) = 1680/752

y así sucesivamente ...

1680/752 está dentro de 0.002 de √5

ALTERNATIVAS PARA NÚMEROS GRANDES

Para los números grandes √L, usar el método de Newton o dividir el número grande en números pequeños y use Bhaskara-Brouncker.

Método de Newton:

Xnew = 1/2 (Xold + L / Xold)

Ejemplo. Encuentre √545678 utilizando el método de Newton:

Xnew ..... Xold

739.7700 ... 700.0000 ..... estimación inicial de 700 para √545678

738.7010 ... 739.7700

738.7002 ... 738.7010

738.7002 ... 738.7002

Ejemplo. Encuentra √4913 usando el algoritmo Bhaskara-Brouncker.

√4913 = 17√17

√17 ≈ 3539/859 ..... de Bhaskara-Brouncker como arriba

√4913 ≈ 17 (3539/859)

≈ 60163/859

que está dentro de 0.06 de √4913

TEOREMA DE GAUSS

Existe una raíz primitiva de un módulo m si y sólo si m son 2, 4 una potencia de un primo impar, o los dos una potencia de un primo impar

TEST DE PRIMALIDAD

es un algoritmo que, dado un número de entrada n, no consigue verificar la hipótesis de un teorema cuya conclusión es que n es compuesto.

La primera técnica fue la criba de eratóstenes (es suficiente con iterar hasta los divisores primos de n menores que n^0.5), teorema de fermat, enunciados de Mersenne (números de mersenne), est de Lucas-Lehmer para números de Mersenne, teorema de Pocklington, teorema de Pepín, Test de Solovay-Strassenm, etc.

PROPIEDADES DE LAS RAÍCES PRIMITIVAS

- Si g es una raíz primitiva módulo n, entonces todo entero que es coprimo con n es congruente con g^i para algún exponente i entre 1 y $\varphi(n)$, inclusive
- Si b tiene orden e mod n, entonces el orden de b^i es $lcm(e,i)/i = e/gcd(e,i)$
- Si hay una raíz primitiva mod n, si d divide $\varphi(n)$ y b es coprimo con n, entonces $b^{\varphi(n)/d} = 1 \; mod \; n$, si y sólo si b es perfecto $d \; th$ potencia módulo n