Premios del Departamento de Matemáticas de la Universidad Autónoma de Madrid para Estudiantes de Secundaria

Tercera Edición, 2008/2009

TRABAJO: Una simulación simplificada de la criptografía

GANADOR EN LA CATEGORÍA DE BACHILLERATO

AUTORES:

- Alicia Ayuso Solís
- Paula Goicoechea Núñez
- Mónica Pérez Serrabona
- Helena Puchades Guerra
- Marcos Torres López

TUTORES:

- Josefa Ranchal Migallón
- Enrique Romero Ruiz del Portal

CENTRO: Colegio Los Sauces (Torrelodones, Madrid)

AUTÓNOMA 40s

FECYT
FUNDACIÓN ESPAÑOLA PARA LA CIENCIA Y LA TECNOLOGÍA

Fundación Vodafone España

Obra Social CosmoCaixa

Una Simulación Simplificada De La Criptografía.

Los Primos De Fermat

1.-INTRODUCCIÓN

"Una pareja de científicos norteamericanos, los doctores Curtis Cooper y Steven Boone, de la Universidad Estatal Central de Missouri, Estados Unidos, flamantes descubridores **del mayor número primo conocido hasta hoy, y compuesto por 9.152.052 cifras**." Esta inocente noticia es la culpable de nuestro trabajo.

Un día, en clase, un compañero comentó que había leído esta noticia y se interesó en cómo se hacía para poder descubrir estas cifras y preguntó acerca de la importancia que podía tener esto para que lo publicaran como noticia. Todos nos reímos, "céntrate en las derivadas" le dijo uno. Para nuestra sorpresa, nuestra profesora no sólo le hizo caso sino que nos "soltó" una charla acerca de los números primos que nos dejó un tanto fríos. No podíamos imaginar que tuvieran ninguna importancia para alguien que no fuera un "friki" de las matemáticas. Ahí quedó todo Al cabo de unos días los profesores nos hablaron de éste concurso de matemáticas y decidimos presentarnos aunque sin saber todavía, incautos de nosotros, acerca de qué íbamos a trabajar. Nuestro primer trabajo consistiría precisamente en buscar un tema que nos llamara la atención. En un principio no se nos ocurría nada como era de esperar. Los profesores nos propusieron una serie de temas entre los que se encontraban las aplicaciones de los números primos. Como nos sonaba al menos de algo, decidimos investigar un poco en Internet y vimos que la criptografía era una de estas aplicaciones y que tenía utilidades muy habituales en la vida diaria, así que mejor que hacer una compilación de aplicaciones decidimos centrarnos en este tema y no buscar más. Según los profesores era la mejor manera de que obtuviéramos un mayor provecho matemático del trabajo.
Criptografía, algo nuevo para todos nosotros y para nuestros profesores por lo que el primer paso fue ponernos como locos a buscar información. Todo nos parecía chino, no entendíamos nada y casi llegamos a arrepentirnos del lío en que nos metíamos antes de empezar, pero al fin y al cabo nos gustan las matemáticas y le echamos muchas ganas.

Empezamos a encontrar y leer (al principio sin entender) un montón de artículos y blogs de diferentes profesores casi todos universitarios y a mandarles mensajes para ver si nos echaban una mano para encontrar cosas sencillas por donde comenzar a crecer. Se quedaban alucinados cuando les decíamos que teníamos 16 años y les preguntábamos sobre lo que habían escrito. Al final nos referiremos a ellos pero debemos decir que nos ayudaron bastante y que se tomaron bastantes molestias por nosotros. Empezamos a buscar mecanismos informáticos que nos dieran idea del funcionamiento real de las cosas. Buscamos ejercicios acerca de cómo se produce el proceso y esto último sí que fue, al principio, un caos donde sólo veíamos números y no había por donde cogerlos. Parece que todo fue malo pero no; debemos reconocer que no tardamos demasiado en ir cogiéndole el aire al tema y que a pesar de nuestras limitaciones íbamos disfrutando cada vez que solucionábamos uno de los obstáculos que se nos presentaban. Es cierto que no hemos conseguido acceder a todo lo que nos propusimos y que los profesores nos tuvieron que echar una mano para comprender ciertas cosas pero lo hemos pasado muy bien realizando el trabajo.

A lo largo de la historia, las aplicaciones de la **criptografía** han sido abundantes. Posiblemente, el primer criptosistema que se conoce fuera documentado por el historiador griego Polibio: un sistema de sustitución basado en la posición de las letras en una tabla. También los romanos utilizaron sistemas de sustitución, siendo el método actualmente conocido como César, porque supuestamente Julio César lo utilizó en sus campañas Otro de los métodos criptográficos utilizados por los griegos fue la escitala espartana, un método de trasposición basado en un cilindro que servía como clave en el que se enrollaba el mensaje para poder cifrar y descifrar. En 1465 el italiano Leon Battista Alberti inventó un nuevo sistema de sustitución polialfabética que supuso un gran avance de la época. Otro de los criptógrafos más importantes del siglo XVI fue el francés Blaise de Vigenere que escribió un importante tratado sobre "la escritura secreta" y que diseñó una cifra que ha llegado a nuestros días asociada a su nombre. Desde el siglo XIX y hasta la Segunda Guerra Mundial las figuras más importantes fueron la del holandés Auguste Kerckhoffs y la del prusiano Friedrich Kasiski. A partir del siglo XX, la criptografía usa una nueva herramienta que permitirá conseguir mejores y más seguras cifras: las máquinas de cálculo, la más conocida puede que sea la máquina alemana Enigma: una máquina de rotores que

automatizaba considerablemente los cálculos que era necesario realizar para las operaciones de cifrado y descifrado de mensajes.
A mediados de los años 70 el Departamento de Normas y Estándares norteamericano publica el primer diseño lógico de un cifrador que estaría llamado a ser el principal sistema criptográfico de finales de siglo: el Estándar de Cifrado de Datos. En esas mismas fechas ya se empezaba a gestar lo que sería la, hasta ahora, última revolución de la criptografía teórica y práctica: los sistemas asimétricos. Estos sistemas supusieron un salto cualitativo importante ya que permitieron introducir la criptografía en otros campos como el de la firma digital.
La era de la criptografía moderna comienza realmente con Claude Shannon, que podría decirse que es el padre de la criptografía matemática. En 1949 publicó el artículo *Communication Theory of Secrecy Systems* y poco después el libro *Mathematical Theory of Communication.*

2.- **OBJETIVOS**:

Desde que el hombre ha aprendido a comunicarse con textos escritos, también ha sentido la necesidad de considerar algunos como privados. Para ocultar este tipo de mensajes, se ha empleando y se sigue usando la criptografía. En este trabajo hemos pretendido que nuestros alumnos se aproximen a este campo que por otra parte nos rodea en nuestra vida cotidiana, desde el uso de firmas digitales, acceso a redes mediante el usuario y password, en las transacciones comerciales vía intranet o en la propia información registrada en nuestra tarjeta de crédito.
Durante el proceso de investigación para la elaboración del presente trabajo, nuestros objetivos serán:

- Familiarizarse con aspectos de la Teoría de Números que están implicados en la explicación y desarrollo del algoritmo RSA, usado en ejemplos de codificación y descodificación así como en todos los procesos intermedios.
- Utilizar el software científico específico tanto en criptografía como para la ejecución de cálculos de considerable envergadura.
- Localizar y manejar bibliografía tanto en inglés como castellano, para profundizar en las teorías matemáticas que el desarrollo del trabajo sacaba a relucir.
- Conseguir llegar a una simulación sencilla con el cifrado y descifrado de un mensaje al final del estudio, utilizando todo lo aprendido con anterioridad.

3.- **CRIPTOGRAFÍA RSA**:

El sistema criptográfico de clave pública RSA es un algoritmo asimétrico cifrador de bloques, que utiliza una clave pública, que se distribuye, y otra privada, que es guardada en secreto por su propietario.
Una clave es un número de gran tamaño, que una persona puede conceptualizar como un mensaje digital, como un archivo binario o como una cadena de bits o bytes.
Cuando se quiere enviar un mensaje, el emisor busca la clave pública de cifrado del receptor, cifra su mensaje con esa clave, y una vez que el mensaje cifrado llega al receptor, éste se ocupa de descifrarlo usando su clave oculta.
Los mensajes enviados usando el algoritmo RSA se representan mediante números y el funcionamiento se basa en el producto de dos números primos grandes.
RSA es una función computacionalmente segura, ya que si bien realizar la exponenciación modular es fácil, su operación inversa, la extracción de raíces de módulo Ø no es factible a menos que se conozca la factorización de e, clave privada del sistema.

Actualmente el método de factorización más rápido conocido es la Criba Numérica Especial de Campo (Special Number Field Sieve): **Rivest**, uno de los coautores de RSA, propuso en 1977 que se intentara factorizar un número de 129 dígitos. Estableció que harían falta alrededor de $4 \cdot 10^{16}$ años de computación para lograrlo, según el estado de la Teoría de Números y la disponibilidad de algoritmos de entonces. Sin embargo, la factorización del número propuesto por Rivest, el famoso RSA-129, se logró el 2 de abril de 1994, después de menos de 8 meses de trabajo, por el desarrollo del método conocido como la criba cuadrática (QS, Quadratic Sieve).
La idea básica de la familia de algoritmos cuadráticos que emplea la criba cuadrática es la siguiente. Siendo n el entero a factorizar, supongamos que x, y son dos enteros tales que $x^2 \equiv y^2 \pmod{n}$. Entonces que $x^2 - y^2 = (x + y)(x - y) \equiv 0 \pmod{n}$, por lo que, si $x - y < n$, resulta que $\text{mcd}(x - y, n) \neq 1$ ha de ser un factor de n.

A modo de ejemplo, tratemos de factorizar con este método el número 91. Después de algunos ensayos, nos daremos cuenta de que $10^2 \equiv 9 \pmod{91}$ y $3^2 \equiv 9 \pmod{91}$; por otro lado, $10 \equiv 10 \pmod{91}$ y $3 \equiv 3 \pmod{91}$, luego x = 10 e y = 3. Como 10 – 3 < 91, tomamos
mcd (10 – 3, 91) = 7, que es un factor de 91.
Conseguida la factorización del número RSA-129, se abordó la factorización del siguiente número:
RSA-130 = 18070 82088 68740 48059 51656 16440 59055 66278 10251
67694 01349 17012 70214 50056 66254 02440 48387 34112
75908 12303 37178 18879 66563 18201 32148 80557
En este caso se hizo uso de una extensión de la criba cuadrática establecida anteriormente llamada la criba del cuerpo de números (SNFS, Special Number Field Sieve)

4.- NÚMEROS PRIMOS:

Un número primo es un entero positivo mayor que 1 que es divisible, solamente, por sí mismo y la unidad. Además, todo entero positivo mayor que uno puede ser escrito de forma única como el producto de primos.
Un dato básico sobre los números primos es que hay infinitos, y como todo en las matemáticas, tiene una demostración numérica, que procedemos a explicar:
Por reducción al absurdo. Supóngase que sólo hay un número finito de números primos y que se definen como ***a, b, c, … , d***. Este conjunto suponemos que los contiene *todos*.
Si multiplicamos estos números primos unos por otros y le sumamos 1 al producto obtenemos un nuevo número:

$$N = (a \cdot b \cdot c \cdot \ldots \cdot d) + 1$$

Obviamente, N es mayor que cualquiera de los números primos individuales **a, b, c, … , d**, y por tanto **N** es diferente de todos ellos. Puesto que estos números son los únicos números primos existentes, concluimos que **N** no es un número primo.
Esto significa que **N** debe ser un número compuesto, es decir, tiene uno o más divisores distintos a 1 y a sí mismo y, por tanto tiene un divisor primo. Puesto que hemos supuesto que **a, b, c, … , d**, constituyen todos los números primos, este divisor primo de N debe estar en algún lugar entre ellos. Dicho de otra manera, **N** es un múltiplo de uno de los números primos **a, b, c, … , d**. Poco importa de cuál de ellos es, pero supongamos que **N** es un múltiplo de **c**. Así, el producto **a·b·c·…·d** es también un múltiplo de **c** ya que aparece como uno de los factores. Pero la diferencia entre **N** y **a·b·c·…·d** será también un múltiplo de **c**. Pero, por definición, **N** es exactamente 1 más que este producto, luego la diferencia es 1.

N tiene que ser divisible por **c.**
a·b·c·…·d tiene que ser divisible por **c**.
} Por lo que **N** - **(a·b·c·…·d)** también tiene que ser divisible por **c**, pero el resultado es 1.

Por tanto, llegamos a la conclusión de que 1 es múltiplo de **c** (o de cualquier otro número primo que es un factor de **N**). Esto claramente, es imposible. Por tanto concluimos que hay infinitos números primos. Después de entender la definición de los números primos, investigamos sobre la relación de primalidad que hay entre los números, chocándonos de bruces con otro nuevo concepto, los números coprimos, por lo que otra vez tuvimos que empezar de cero e investigarlos:

4.a.- Números coprimos:

Dos números enteros son coprimos o primos entre sí si su máximo común divisor (mcd) es 1. En tal caso los únicos divisores comunes son -1 y 1.
Esto equivale a decir que la fracción es irreducible (no se puede simplificar).

- 7 y 11 si son coprimos porque son ambos primos.
- En el caso de 11 y 49 también son coprimos por que 11 es primo y 49 no es múltiplo de 11.
- Coprimos en general significa primos entre sí

Propiedades elementales de los coprimos

Si *a* y *b* son coprimos entonces su mínimo común múltiplo (mcm) es el producto *a·b*, que es también el menor denominador común de las fracciones irreducible con denominador a y b.

Por ejemplo: $\frac{3}{7}-\frac{5}{8}=\frac{3\cdot 8-5\cdot 7}{7\cdot 8}=-\frac{11}{56}$

Esto es consecuencia de la relación entre el mcm *m* y el mcd *d*: **m·d = a·b**.

Si *a* y *b* son coprimos, entonces lo serán también *a* y *a + b*, *a* y *a - b* y más generalmente *a* con $a+k\cdot b$, con $k\in Z$.

En particular *a* será coprimo con *r*, el resto de la división euclidiana de *a* por *b*. Este hecho, fundamental, es la base del algoritmo de Euclides, el método más rápido de hallar el máximo común divisor y por lo tanto de saber si dos enteros son o no coprimos.

Si se escogen al azar, la probabilidad de que sí lo sean es de $\frac{6}{\pi^2}$

4. b.- Función □ de Euler

La función □(n) nos da la cantidad de enteros positivos menores o iguales que n que son primos respecto a n.
Es una función multiplicativa condicional: si *m* y *n* son primos entre sí, entonces
□(*mn*) = □(*m*) □(*n*).
Puede definirse como: $\varphi(n) = cardinal\ de\{n \in N / n < m\ y\ mcd\{m,n\} = 1\}$
Ejemplo: $\varphi(6) = 2$, porque existen 6 enteros positivos menores o iguales que n, a saber, 1,2,3,4,5,6, de los cuales solo dos, el 1 y el 5 cumplen que mcd(1,6)=1 y mcd(5,6)=1.

Nos interesa saber si es posible calcular dicha función, sin tener que ir probando con todos los enteros menores o iguales que n, ya que eso sería una tarea muy laboriosa.
Si conocemos la factorización en factores primos de n, podemos calcular con suma facilidad el valor de $\varphi(n)$.
En precisamente que para cualquier k entero positivo estrictamente menor que p, se cumplirá que mcd (k,p)=1. Por lo tanto $\varphi(p) = p - 1$.
Ejemplo: $\varphi(7) = 7 - 1 = 6$ ó $\varphi(13) = 13 - 1 = 12$

-Sea ahora $n = p^m$, es decir n es una potencia de algún número primo. De los números k que son enteros positivos menores o iguales que p^m, tenemos que sólo la unidad y los múltiplos de p tienen algún divisor común con p^m. Ello quiere decir que existirán p^{m-1} números con algún divisor en común con p diferente de la unidad. Por lo tanto existirán $(p-1)\cdot p^{m-1}$ números tales que mcd (k, p^m)=1. Es decir tendremos que $\varphi(p^m) = (p-1)\cdot p^{m-1}$.
Ejemplo: $\varphi(8) = \varphi(2^3) = (2-1)\cdot 2^2 = 4$ ó $\varphi(27) = \varphi(3^3) = (3-1)\cdot 3^2 = 18$

-Siguiendo consideraciones análogas, veríamos que si $n = p^m \cdot q^s$ con p y q primos, entonces $\varphi(n) = (p-1)\cdot p^{m-1} \cdot (q-1)\cdot q^{s-1}$ y en general si $n = {p_1}^{e_1} \cdot {p_2}^{e_2} \cdot \ldots \cdot {p_n}^{e_n}$, siendo $p_1, p_2, \ldots p_n$ primos, tendríamos que
$\varphi(n) = (p_1 - 1)\cdot {p_1}^{e_1 - 1} \cdot (p_2 - 1)\cdot {p_2}^{e_2 - 1} \cdot \ldots \cdot (p_n - 1)\cdot {p_n}^{e_n - 1}$.
Ejemplo: $\varphi(18) = \varphi(2 \cdot 3^2) = (2-1)\cdot(3-1)\cdot 3 = 6$ ó
$\varphi(118125) = \varphi(3^3 \cdot 5^4 \cdot 7) = (3-1)\cdot 3^2 \cdot (5-1)\cdot 5^3 \cdot (7-1) = 54000$

Un ejemplo de uso:

φ de 9,10 y 11:
$\varphi(9) = 6$ puesto que 1, 2, 4, 5, 7, y 8 son primos con 9.
$\varphi(10) = 4$ puesto que 1, 3, 7 y 9 son coprimos con 10.
$\varphi(11) = 10$ puesto que todos los números menores que un primo son coprimos con él.

Para todo natural n>1 φ(n) es el número de elementos invertibles del anillo cíclico Z/Z_n. Aunque Z/Z_1 no es propiamente un anillo (sus dos leyes «+» y «×» se confunden), tiene un único elemento invertible, y por tanto se extiende Φ con Φ(1) = 1. No se define Φ(0). Los invertibles de Z_n corresponden a los m tales que *m* es coprimo con *n*; con $0 < m \leq n$.

Miremos los primeros valores de □:

n	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35
□(n)	1	1	2	2	4	2	6	4	6	4	10	4	12	6	8	8	16	6	18	8	12	10	22	8	20	12	26	12	28	8	30	16	20	16	24

No se percibe ningún orden subyacente, lo que no es de extrañar porque esta función depende de la descomposición en primos, que es muy irregular.

Una vez definimos números primos y coprimos, seguimos investigando sobre cómo comprobar la primalidad de un número y cómo generarlos, pero para ello necesitamos comprender la aritmética modular.

5.- ARITMÉTICA MODULAR:

La aritmética modular es un sistema aritmético para clases de equivalencia de números enteros llamadas clases de congruencia. Algunas veces se le llama, sugerentemente, aritmética del reloj, ya que los números 'dan la vuelta' tras alcanzar cierto valor (el módulo).

La aritmética modular puede ser construida matemáticamente mediante la relación de congruencia entre enteros; esto quiere decir que dos números enteros *a* y *b* tienen el mismo resto al dividirlos por un número natural *m*, llamado el **módulo**; esto se expresa utilizando la notación $a \equiv b(\mathrm{mod}\, m)$ que se expresa diciendo que *a es congruente* con *b* módulo *m*. Las siguientes expresiones son equivalentes:

- *a* Es congruente con *b* módulo *m*

  $$a \equiv b(\mathrm{mod}\, m)$$

- El resto de *a* entre *m* es el resto de *b* entre *m*

  $$a \bmod m = b \bmod m$$

- *m* divide exactamente a la diferencia de *a* y *b*

  $$m/(a-b)$$

- *a* se puede escribir como la suma de *b* y un múltiplo de *m*

  $$\exists k \in \mathrm{Z} \quad a = b + km$$

Esta relación de congruencia es compatible con las operaciones en el anillo de enteros: suma, resta, y multiplicación. En álgebra, un **anillo** es una estructura algebraica formada por un conjunto y dos operaciones que están relacionadas entre sí mediante la propiedad distributiva.

Para un determinado módulo n, ésta se define de la siguiente manera:

a y b se encuentran en la misma "clase de congruencia" módulo n, si ambos dejan el mismo resto si los dividimos por n, o, equivalentemente, si a □ b es un múltiplo de n.

Todo esto parece muy difícil así escrito, pero se entiende muy fácilmente con un ejemplo:

Ejemplo: 14 y 26 son congruentes en módulo 12,

Ya que dejan el mismo resto al dividirlos entre 12;(2), ó también 14-26=-12, que es múltiplo de 12.

$$a \equiv b(\mathrm{mod}\, n)$$

Otra forma de escribir módulo de n es:

Z_n={0, 1, 2, ⋯, n-1} subconjunto de Z, conjunto de números enteros entre 0 y n-1

Aplicando los conceptos anteriores a números concretos, tendremos:

Z_{12} ={0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11}

$Z^*_{n:}$ conjunto que incluye todos los números menores que n y coprimos con él.

Z^*_{12} ={1, 2, 4, 7, 8, 11}

Si n es un número primo, entonces ambos conjuntos coincidirían, salvo el 0, que no se incluye en el segundo conjunto. El grado de organización que formaría sería un cuerpo, no un anillo, y al ser todos los elementos coprimos con él; todos tendrían inverso.

Z_{13} ={0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12}

Z^*_{13} ={1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12}

Una vez que tenemos la base, ya podemos profundizar en cómo saber si un número es primo o no, utilizando distintos criterios de primalidad, algunos de fácil comprensión y otros en los que nos tuvimos que poner a pensar:

Criterios de primalidad:

-Criba de Eratóstenes: es una manera sencilla de hallar todos los números primos menores o iguales que un número dado. Se basa en confeccionar una lista de todos los números naturales desde el 2 hasta ese número y tachar repetidamente los múltiplos de los números primos ya descubiertos. Este método sólo sirve para números pequeños, por lo que no nos es útil.

- Test de primalidad de Fermat: este criterio se basa en el "Pequeño teorema de Fermat", que explicamos brevemente a continuación:

-Si *p* es un número primo, entonces, para cualquier número natural *a*:

$a^p \equiv a(\mathrm{mod}\, p)$ Todas estas fórmulas las comprendimos gracias a nuestra anterior labor de investigación con la aritmética modular.

Aunque son equivalentes, el teorema suele ser representado de la siguiente forma:

-Si *p* es un número primo, entonces, para cada número natural *a* coprimo con *p*:

$a^{p-1} \equiv 1(\mathrm{mod}\, p)$

Esto quiere decir que si se eleva un número *a* a la *p*-ésima potencia y al resultado se le resta *a*, lo que queda es divisible por *p*.

Aunque se profundizará más adelante, dos números son, como ya vimos, coprimos si, por definición, no tienen ningún factor primo en común.

De este teorema se deduce el criterio de primalidad de Fermat, que es el siguiente:

Un número *p* será primo si cumple la siguiente igualdad: $\frac{a^p - a}{p} = \mathrm{Z}$.

-Si la ecuación se cumple, *p* es primo
-Si la ecuación no se cumple, *p* es compuesto

-Test de Lucas-Lehmer: este test se aplica solo a los números de Mersenne sobre los que profundizaremos después.

Si existe un número natural *a* menor que *n* y mayor que 1 que verifica las condiciones $a^{n-1} \equiv 1 (\mathrm{mod}\, n)$, así como $a^{(n-1)/q} \neq 1 (\mathrm{mod}\, n)$ para todos los factores primos *q* de *n* □ 1, entonces *n* es primo. Si no puede encontrarse tal *a*, entonces *n* es un número compuesto.

Ejemplo: $n = 71; n-1 = 7 \cdot 5 \cdot 2; \quad a = 11$ (tiene que ser coprimo con n) $11^{70} \equiv 1 (\mathrm{mod}\, 71)$

Esto no demuestra que el orden multiplicativo de 11 mod 71 es 70, porque algún factor de 70 aún podría funcionar arriba. Verificamos entonces 70 dividido por sus factores primos:

$11^{35} = 70 \neq 1 (\mathrm{mod}\, 71)$

$11^{14} = 54 \neq 1 (\mathrm{mod}\, 71)$

$11^{10} = 32 \neq 1 (\mathrm{mod}\, 71)$

Entonces, el orden multiplicativo de 11 mod 71 es 70 y de esta manera, 71 es primo.

Para encriptar de forma segura, se usan números primos muy grandes, cuyos ejemplos encontraremos al final del trabajo, pero para generarlos hay que recurrir a algunos polinomios, que tienen, por desgracia, ciertas limitaciones. Los más importantes son los siguientes.

-Polinomio de Euler: $x^2 + x + 41$.

Este sencillo polinomio genera números primos para todos los valores de x entre 0 y 39. El polinomio $x^2 - x + 41$ también genera números primos para todos los valores de x entre 1 y 40.

-Polinomios de Legendre: $2x^2 + 29$, genera números primos para valores de x entre 0 y 28.
$x^2 + x + 17$ genera números primos para valores de x entre 0 y 15.

El polinomio de Euler puede transformarse, haciendo x = y - 40 en este otro polinomio: $y^2 - 79y + 1601$ que genera números primos para 80 números consecutivos.

Goldbach demostró que ningún polinomio puede generar números primos para todos los valores y Legendre que ninguna función algebraica racional genera siempre números primos.

Siguiendo el análisis de los números primos, ahora toca nombrar los tipos que hay:

-Gemelos: *p* y *p+2* lo son si son los dos primos.
-De Mersenne: son los de forma $M_p = 2^p - 1$, donde *p* es primo.
-De Sophie Germain: dado *p* primo, es de Sophie Germain si $2p + 1$ también es primo.
-De Fermat: son los números de la forma $2^{2^n} + 1$, con *n* natural. Los únicos números primos de Fermat que se conocen hasta la fecha son los cinco que ya conocía el propio Fermat, correspondientes a *n* = 0, 1, 2, 3 y 4. Por ahora, todos los demás a partir de *n* = 5 han resultado ser compuestos.
-Fuertes: estos son los más importantes para la criptografía, ya que son los utilizados en el sistema de clave pública RSA. En este método se seleccionan dos números primos *p* y *q* suficientemente grandes (de centenas de dígitos) y se obtiene su producto *n=pq*. Podemos hacer público el número n (sería la clave pública) porque es muy difícil obtener los factores *p* y *q*.

Para hacer más difícil la descomposición en factores primos del número *n*, se eligen *p* y *q* de tal forma que cumplan las siguientes condiciones:

-mcd ((p-1), (q-1)) debe ser pequeño.
- p-1 y q-1 deben tener algún factor primo grande p' y q'.
- Tanto p'-1 como q'-1 deben tener factores primos grandes.
- Tanto p'+1 como q'+1 deben tener factores primos grandes.
- Las dos primeras condiciones se cumplen si tanto (p-1)/2 como (q-1)/2 son primos.

Los números p y q que cumplan estas condiciones, se denominaran primos fuertes.
Otra característica muy importante de los números primos es que no están distribuidos de una forma regular, lo único que está claro es que a medida que avanzamos en orden creciente, va disminuyendo el número de ellos, lo cual tiene lógica, ya que a medida que los números son más grandes, hay más posibles factores de éstos. Pero cuando los números primos son muy grandes, si que siguen un cierto patrón, que se expone en el TEOREMA DE LOS NÚMEROS PRIMOS:
π (x) representa el número de primos menores o iguales que el número entero x.
x/π (x) representa el inverso de π (x).

x	π (x)	π (x) / x	r(x) = x /ln (x)
10	4	0,40000000	2,50000000
100	25	0,25000000	4,00000000
1.000	168	0,16800000	5,95238095
10.000	1.229	0,12290000	8,13669650
100.000	9.592	0,09592000	10,4253545
1.000.000	78.498	0,07849800	12,7391781
10.000.000	664.579	0,06645790	15,0471201
100.000.000	5.761.455	0,05761455	17,3567267
1.000.000.000	50.847.534	0,05084753	19,6666387
10.000.000.000	455.052.512	0,04550525	21,9754863

En la tabla se aprecia que en la fila de x/π (x), cuando los números son grandes, la diferencia aproximada es de 2.3. Sabiendo que ln (10)= 2.30258... Gauss formuló la conjetura de que π (*n*) es aproximadamente igual a *n*/*Ln* (*n*). π (n) $\approx$ n / ln n para valores grandes de a.
Un ejemplo para demostrar la conjetura de Gauss ayudándonos de la tabla sería:

X= 1.000.000.000, x / ln(x)= 1.000.000.000 / ln (1.000.000.000) =
= 48.254.942 $\approx$ 50.847534
□(x) =50.847.534

A medida que los números vayan siendo mayores, las cifras se irán igualando cada vez más.
Una vez que ya hemos finalizado la investigación sobre los números primos y coprimos, pasamos a profundizar en la aritmética modular con el cálculo de inversos:

Cálculo de inversos:

¿Inversos?... ¿cuál es el inverso de un número? El inverso de un número es otro número que multiplicado por el primero de 1.

El multiplicador modular inverso de un entero n módulo p es un entero m tal que $n^{-1} \equiv m(\mathrm{mod}\ p)$

Esto significa que es el multiplicador inverso en el anillo de los enteros módulo p. Es equivalente a $mn = 1(\mathrm{mod}\ p)$

El multiplicador inverso de n módulo p existe si y sólo si n y p son coprimos, es decir, si MCD(n, p)=1.

Para calcular inversos podemos seguir distintos caminos (que**,** sinceramente, nos pareció a cada cual más difícil…)

6.- ALGORITMO EXTENDIDO DE EUCLIDES

Una de las muchas aplicaciones de este algoritmo son los inversos modulares:

En criptografía deberá estar permitido invertir una operación para recuperar un cifrado ⇒ descifrar.

Aunque la cifra es una función, en lenguaje coloquial la operación de cifrado podría interpretarse como una "multiplicación" y la operación de descifrado como una "división", si bien hablaremos en este caso de una multiplicación por el inverso.

La analogía anterior sólo será válida en el cuerpo de los enteros Zn con inverso.

-Luego, si en una operación de cifrado la función es el valor a dentro de un cuerpo n, deberemos encontrar el inverso a-1 mod n para descifrar; en otras palabras…hallar el número que multiplicado por a nos de 1 (siempre en módulo de n)

Si $a \cdot x \equiv 1 \bmod n$

Se dice que x es el inverso multiplicativo de a en Zn y se denotará por a^{-1} **.**

-Si no hay primalidad entre a y n, es decir, Si mcd (a, n) ≠ 1, no existe el inverso. Por ejemplo, si n = 6, en no existe el inverso del 2, pues la ecuación 2∗x ≡ 1 mod 6 no tiene solución.

-Si n es un número primo p, entonces todos los elementos de Zp salvo el cero tienen inverso. Por ejemplo, en Z_5 se tiene que:

$$1 - 1 \bmod 5 = 1; \quad 2 - 1 \bmod 5 = 3, \quad 3 - 1 \bmod 5 = 2; \quad 4 - 1 \bmod 5 = 4$$

Bueno, ya sabemos qué es un inverso, pero, ¿cómo se calcula?:

Inverso multiplicativo:

*Hay una propiedad que dice que un número a tiene inversa módulo n, si no existe ningún número (excepto 1) menor que a y menor que n que los divida de forma exacta. Esto es a lo que se llama primos relativos. 8 y 5 serían primos relativos, porque no hay ningún número que los divida, aunque 8 no sea primo. Su máximo común divisor es 1.

En el ejemplo del módulo 7, vemos que todos los números (el cero no cuenta) tienen que tener inversa, porque 7 es primo absoluto y no va a existir ningún número que lo divida.

* La inversa del 1 es el 1: 1 * 1 = 1 que dividido entre 7 es igual a cero y resto 1, $(1 \cdot 1 \bmod 7 = 1)$

- La inversa del 2 es el 4: 2 * 4 = 8 que dividido entre 7 es igual a 1 y de resto 0, $(2 \cdot 4 \bmod 7 = 1)$ etc...

Ejemplo:

Sabemos que en el módulo 7 todo su conjunto de números, menos el cero, tienen inversa, porque 7 es un número primo y nos lo dice la propiedad anterior. Esto quiere decir que hay 6 números (del 1 al 6) que tienen inversa, y si $\varphi(7)$ nos dice la cantidad de números que tienen inversa, queda que $\varphi(7) = 6$.

De forma genérica, si n es un número primo, $\varphi(n) = n - 1$. (El menos 1 es porque no contamos con cero).

Por la misma razón, si n está formado por la multiplicación de dos números primos, $n = p \cdot q$, entonces $\varphi(n) = (p-1) \cdot (q-1)$

Conclusión: ¡Euclides ya está dominado!, Pero aún nos queda mucho por aprender…

7.- TEOREMA DE EULER:

Una alternativa al algoritmo euclidiano.

- De acuerdo con el Teorema de Euler, si n es coprimo con p, es decir, si mcd(n,p)=1, entonces, $n^{\varphi(p)} \equiv 1 (\bmod p)$

Esto se deduce del Teorema de Lagrange y del hecho de que n pertenece al grupo multiplicativo de enteros módulo n (Z / Z_p) si y sólo si n es coprimo con p. Así pues, $n^{\varphi(p)-1} \equiv n^{-1} (\bmod p)$

Donde □(p) es la Función □ de Euler.

De esta forma se puede obtener el multiplicador modular inverso de n módulo p de forma directa:

$$n^{\varphi(p)-1} \equiv m (\bmod p)$$

En el caso especial en que p es primo,

$$\varphi(p) = p - 1$$

- Una aplicación del teorema de Euler es la resolución de ecuaciones de congruencia.

Por ejemplo, se desea encontrar todos los números *x* que satisfacen

$$5x \equiv 2 (\bmod 12)$$

En otras palabras, todos los números que al multiplicarlos por 5, dejan residuo 2 en la división por 12. O de otra forma, todos los números *x* tales que 12 divida a 5x-2.

El teorema de Euler dice que

$$5^{\varphi(12)} = 5^4 \equiv 1(\mathrm{mod}12)$$

Por lo que, multiplicando ambos lados de la ecuación por 5^3:

$$5^3 \cdot 5x \equiv 5^3 \cdot 2 = 250 \equiv 10(\mathrm{mod}12)$$
$$5^4 x \equiv 10(\mathrm{mod}12)$$
$$x \equiv 10(\mathrm{mod}12)$$

Entonces, la conclusión es que, cualquier número que al dividirse por 12 tenga residuo 10, será una solución de la ecuación. Se puede verificar con un ejemplo. Si se divide 34 entre 12, el residuo es 10, por lo que *x*=34 debe funcionar como solución. Para verificarlo, se divide 34·5=170 entre 12, obtenemos un cociente 14 y un residuo 2, como se esperaba

8.- ECUACIÓN DIOFÁNTICA:

Definición. El término ecuación diofántica se usa para designar una ecuación en una o más incógnitas que va a ser resuelta en los enteros. La ecuación diofántica más simple es la ecuación diofántica lineal en dos incógnitas $ax + by = c$ donde a y b son enteros dados no ambos cero.

Para que la ecuación tenga solución, es necesario que c sea mcd de a y b.

En el caso en el que el mcd sea 1, la ecuación tiene infinitas soluciones. También hay infinitas soluciones si c es un múltiplo del mcd (a, b).

Si c no es un múltiplo del mcd(a, b), entonces la ecuación diofántica no tendría soluciones.

¿Parecía muy inofensiva?...pues esta ecuación se las trae…

Un criterio para conocer cuando una ecuación diofántica de este tipo: $ax + by = c$, posee solución lo proporciona el siguiente teorema.

- Teorema. La ecuación diofántica $ax + by = c$ tiene solución sí y sólo sí d es múltiplo de c, donde d = M.C.D. (a, b).

Si x_o y y_o es una solución particular de esta ecuación entonces todas las otras soluciones están dadas por: $x = x_0 + (\frac{b}{d}) \cdot t$ $y = y_0 - (\frac{a}{d}) \cdot t$ para t entero arbitrario.

Ejemplo 1:

La ecuación $2x + 10y = 17$ no tiene solución porque: 2 = M.C.D.(2, 10) no divide a 17.

Ejemplo 2:

La ecuación 5x + 6y = 8 tiene solución porque: 1 = M.C.D.(5, 6) divide a 8.

¿Cómo hallamos una solución particular?

Existen dos métodos. El primero es por simple inspección, pero si así no fuera posible, podemos utilizar el algoritmo de Euclides así:

1 = 5x + 6y Se hallan x y y utilizando el algoritmo anteriormente citado.

$6 = 1 \cdot 5 + 1$ y $5 = 5 \cdot 1 + 0$ Luego $1 = 6 - 1 \cdot 5$ .Lo que quiere decir que $1 = 5 \cdot (-1) + 6 \cdot 1$,

Entonces:

x = -1, y = 1.

En la expresión $5 \cdot (-1) + 6 \cdot 1 = 1$ se multiplican ambos miembros por 8 y se obtiene: $5 \cdot (-8) + 6 \cdot (8) = 8$

Luego la solución particular de la ecuación diofántica es de la forma siguiente:

x_0 = -8, y_0 = 8. La solución general será: $\begin{cases} x = (-8) + 6t \\ y = 8 - 5t \end{cases}$

9.- <u>TEOREMA CHINO DEL RESTO:</u>

Nos dice que un sistema del tipo: $\begin{cases} x \equiv a_1 \bmod(m_1) \\ x \equiv a_2 \bmod(m_2) \\ ..... \\ x \equiv a_n \bmod(m_n) \end{cases}$ donde los m_i, m_j son primos entre sí,

admite una solución única en $\bmod(m_1 \cdot m_2 \cdot ... \cdot m_n)$, $[x_0] \in Z / m_1 m_2 .. m_n$. El resto de soluciones sería de la forma $x = x_0 + t \cdot m_1 \cdot m_2 \cdot ... \cdot m_n$

Cuando vimos esto nos desmoralizó un poco y decidimos intentar entender y desgajar la demostración pero sólo para tres ecuaciones. Allá vamos, ¿quién dijo miedo?

Las soluciones se pueden escribir de la forma: $x_0 = a_1 \cdot b_1 \cdot N_1 + a_2 \cdot b_2 \cdot N_2 + a_3 \cdot b_3 \cdot N_3$,

donde $N_1 = m_2 \cdot m_3$, $N_2 = m_1 \cdot m_3$, $N_3 = m_1 \cdot m_2$ y $b_1, b_2\ y\ b_3$ serían los inversos respectivos de $N_1, N_2\ y\ N_3$ en módulo $m_1, m_2\ y\ m_3$ respectivamente, es decir,

$[b_1] \cdot [N_1] = [1] en\ Z / m_1$, $[b_2] \cdot [N_2] = [1] en\ Z / m_2$ y $[b_3] \cdot [N_3] = [1] en\ Z / m_3$

Sabemos que N_1 tiene inverso en módulo m_1 porque es $N_1 = m_2 \cdot m_3$ que son coprimos con m_1.

Por otra parte, sabemos que $x_0\ y\ a_1$ son congruentes en módulo m_1, es decir,

$x_0 = a_1 \bmod(m_1)$ y además $x_0 = a_1 \cdot b_1 \cdot N_1 + a_2 \cdot b_2 \cdot N_2 + a_3 \cdot b_3 \cdot N_3$, que como

$N_2 = m_1 \cdot m_3$, $N_3 = m_1 \cdot m_2$, son múltiplos de $m_1 \Rightarrow [x_0] = [a_1 \cdot b_1 \cdot N_1] en\ Z / m_1$, es decir, son congruentes en módulo m_1.

Esto se pone bonito porque vamos llegando a lo que queremos y además entendiéndolo, cosa que no esperábamos del todo. Resulta que entonces: $\Rightarrow [x_0] = [a_1] \cdot [1] en\ \mathrm{Z}/m_1$ ya que $[b_1] \cdot [N_1] = [1] en\ \mathrm{Z}/m_1$ por ser b_1 el inverso de N_1 en este módulo. Con esto queda demostrado como queríamos que $[x_0] = [a_1] en\ \mathrm{Z}/m_1$ conclusión generalizable $[x_0] = [a_j] en\ \mathrm{Z}/m_j$.

Vamos a ver ahora si realmente entendemos cómo se utiliza para calcular potencias grandes, utilidad que nos ocupará en la criptografía RSA:

Calculemos $x = (25)^{200} \bmod(100)$

Sabemos que: $100 = 4 \cdot 25 \Rightarrow \begin{cases} m_1 = 4 \\ m_2 = 25 \end{cases}$. Evidentemente, $x = (25)^{200} \equiv 0\ en \bmod(25)$ por ser múltiplo de 25. Ahora vamos a calcularlo en módulo 4:

$[25]^{200} = [1]^{200} = [1]\ \ en\ \ \mathrm{Z}/4$ pues bien, entonces: $\begin{cases} x \equiv 1 \bmod(4) \\ x \equiv 0 \bmod(25) \end{cases}$ lógicamente $x = x_0 + t \cdot 100$ y como explicamos antes: $x_0 = 1 \cdot b_1 \cdot N_1 + 0 \cdot b_2 \cdot N_2 = b_1 \cdot 25 + 0 = 25$.

Aquí dos aclaraciones: a) $N_1 = m_2$ por la misma forma de definirlo.

b) b_1 es el inverso de 25 en módulo 4. Como el 25 y el 1 son el mismo número en módulo 4, $b_1 = 1\ ó\ 25 \Rightarrow x = (25)^{200} \bmod(100) \Rightarrow x = 25 + t \cdot 100$. Somos grandes.

Una vez asimilados los conceptos matemáticos, los aplicamos en la criptografía a través del algoritmo RSA:

10.- ALGORITMO RSA:

En febrero de 1978 Ron Rivest, Adi Shamir y Leonard Adleman proponen un algoritmo de cifra de clave pública llamado RSA. Este algoritmo se basa en la criptografía asimétrica, el cual es un método criptográfico que usa un par de claves para el envío de mensajes. Ambas claves pertenecen a la misma persona a la que se le ha enviado el mensaje, aquí intervienen dos claves, una pública, que se puede entregar a cualquier persona y otra privada, la cual el propietario debe mantener en secreto. De esta forma el remitente usa la clave pública del destinatario para cifrar el mensaje a enviar, pero una vez cifrado, únicamente la clave privada del destinatario podrá descifrarlo.

Los pasos del algoritmo son los siguientes:

- Se eligen dos números primos p y q.
- Se multiplican esos dos números, dando como resultado n.
- Se calcula $\phi(n) = (p-1)(q-1)$, que es la función de Euler, explicada anteriormente.
- Se elige una clave pública e de forma que $1 < e < \phi(n)$ y que cumpla con la condición: mcd $[e, \phi(n)] = 1$.
- Se calcula la clave privada d = inv $[e,\phi(n)]$. para el cálculo de este inverso se pueden usar indistintamente los distintos métodos anteriormente explicados
- Se hace pública la pareja (e, n).
- Se guarda en secreto la clave d.

La operación de cifrado consiste en: $C = M^e \bmod(n)$

La operación de descifrado consiste en: $D = C^d \bmod(n)$

Recuperamos M, teniendo en cuenta que $e \cdot d = 1 + k \cdot \phi(n)$ ya que:

$(M^e)^d = M^{e \cdot d} = M^{1 + k \cdot \phi(n)} = M \cdot M^{\phi(n)} \cdot \cdot M^{\phi(n)} = M$ ya que todos estos factores $M^{\phi(n)} = 1$, obviamente.

Uno de los problemas del descifrado es que la clave d suele ser muy grande, por lo que la exponenciación es muy costosa, para solucionar esto se recurre al teorema del resto chino, que ya ha sido explicado en la sección de aritmética modular.
Un ejemplo del método RSA es el siguiente:
- p = 7, q = 23
- n = p * q = 7 * 23 = 161
- $\phi(n)$ = (p-1) (q-1) = (7-1) * (23 -1) = 132
- e = 5
- e * d mod n = 1 => **d = 53**, lo calcularemos de las siguientes formas; en la primera de ellas hemos usado el teorema de Euler.

$$d \rightarrow e^{f(f(n))-1} \bmod f(n) = 1 \rightarrow 5^{40-1} \bmod 132 = \mathbf{53}$$
$$f(f(n)) = 2^{2-1} * (2-1) * 3^{1-1} * (3-1) * 11^{1-1} * (11-1) = 2 * 2 * 10 = 40$$

La segunda forma de calcularlo es mediante la ecuación diofántica:

$$5 * X + 132 * Y = 1$$
$$132 = 5 * 26 + 2 \rightarrow 2 = 132 - 5 * 26$$
$$5 = 2 * 2 + 1 \rightarrow 1 = 5 - 2 * 2 \rightarrow 1 = 5 - 2 * (132 - 5 * 26) \rightarrow$$
$$1 = 5 - 132 * 2 + 5 * 52 \rightarrow 1 = -132 * 2 + 5 * \mathbf{53}$$

- Clave pública = (5, 161)
- Clave privada = (53, 161)

A partir de estas claves, haremos una operación de cifrado y de descifrado, siendo el mensaje a cifrar (M) = 15, el cifrado es el siguiente:
$15^{5} \bmod 161 = 99$
El número 16 es el mensaje que se enviaría y para descifrar sería como sigue:
$99^{53} \bmod 161 = 15$
Como se puede observar obtenemos el mensaje cifrado al principio.

¿Cuál es la fortaleza de este algoritmo?

El intruso que desee conocer la clave secreta d a partir de los valores n y e se enfrentará al Problema de la Factorización de Números Grandes (PFNG), puesto que la solución para conocer esa clave privada es conocer primero el valor del Indicador de Euler $\phi(n)$ = (p-1)(q-1) para así poder encontrar d = inv [e, $\phi(n)$], pero para ello deberá saber los valores de los primos p y q.
El tamaño que deben de tener los parámetros p y q para que el algoritmo sea suficientemente seguro deben de ser del orden de 500 bits, y que difieran de unas cuantas cifras, otra característica que es favorable a la hora de elegir p y q, es que sean primos seguros, es decir, que al multiplicarlos por 2 y sumarle 1, el resultado sea un numero primo. Con esta última medida nos aseguramos de tener las mínimas claves parejas posibles, que veremos a continuación.
Además la clave publica no debe ser demasiado baja, ni muy alta debido a la dificultad de las operaciones. Generalmente se usa el nº 4 de Fermat: $2^{2^4} + 1 = 65537$.

Claves privadas parejas:
Una clave privada pareja (CPP) permite descifrar el cifrado C, resultado de una cifra con la clave pública e. En el algoritmo RSA habrá como mínimo una CPP. El que existan las CPP se debe a que las claves inversas e y d lo serán en el cuerpo $\phi(n)$, y en cambio la cifra se realiza en el cuerpo n. Esto no compromete en absoluto la fortaleza del sistema, pues para saber las claves parejas se necesita conocer p y q, los cuales solo es posible conocer a través de la factorización de n.
Por ejemplo:
Siendo p = 13, q = 19, n = 247, $\phi(n)$ = 216, elegimos e = 41, d = inv (41, 216) = 137.
El mensaje a cifrar M = 87, por lo tanto el cifrado quedaría: $C = M^{e} \bmod n = 87^{41} \bmod 247 = 159$, y para descifrar haríamos: $M = C^{d} \bmod n = 159^{137} \bmod 247 = 87$, sin embargo también podríamos descifrarlo tomando d = 29, 65, 101, 173, 209 y 245.

Para saber cuántas CPP;

$p = 13, \quad q = 19 \Rightarrow n = 247 \quad y \quad \Phi(n) = 216$

$e = 41 \Rightarrow d = 137$

$Si \quad \gamma = mcm[(p-1),(q-1)] \quad además \quad d = e^{-1} \bmod \gamma = inv(e,\gamma)$

Es decir, d es la inversa de e en módulo y.

La clave pública e tendrá ☐ claves parejas d_i, de la forma:

$$d_i = d_\gamma + i\gamma \quad i = 0,1,....\lambda \qquad 1 < d_i < n \text{ siendo } \lambda = \lfloor (n - d_\gamma)/\gamma \rfloor$$

En nuestro ejemplo:

$\gamma = mcm(12,18) = 36 \Rightarrow d\gamma = inv(41,36) = 29 \Rightarrow d_i = d_\gamma + i\gamma = 29 + i36$ lo que nos lleva a los números 29, 65, 101, 173, 209, 245

$El\ n^o\ sería\ \lambda = [(247 - 29)/36] = [6,05] \Rightarrow 6$

A modo de ejemplo, podemos observar, el tamaño de los números obtenidos, cuando efectuamos los pasos indicado antes, con primos más grandes:

1.- Hemos seleccionado dos números primos p y q, a través de la tabla de números primos que encontramos en el software Expocrip. En nuestro caso p = 2459 y q = 9743
2.- Obtenemos su producto n = pq
3.- Igualmente la función de Euler ϕ(n) = (p-1) (q-1)= 23945836
4.- Seleccionamos e, parte de la clave pública de manera aleatoria en este caso 17 siendo su mcm con ϕ(n) igual a 1.
5.- Calculamos el inverso de e.
6.- Se calcula la clave privada d = inv [e,ϕ(n)].
7.- Obtenemos las parejas de números que constituirán la clave pública y privada.

```
> p := 2459;                                                    p := 2459
> q := 9743;                                                    q := 9743
> n:= p * q;                                                    n := 23958037
> f(n):= (p-1) * (q-1);                                         f(23958037) := 23945836
> e:= 17; #Elegido aleatoriamente, pero que su mínimo común divisor con f(n) sea 1.
                                                                e := 17
> d:= e ^ (-1) mod (f(n)); #Se calcula el inverso de "e", en modulo f(n).
                                                                d := 7042893
> e * d mod (f(n));  #Mediante esta operación vemos que "d" es el inverso de "e".
                                                                1
> (e, n); #Clave pública.
                                                                17, 23958037
> (d, n); #Clave privada.
                                                                7042893, 23958037
> M:= 6447; #Mensaje a enviar.
                                                                M := 6447
> c:= M^e mod (n); #Operación de cifrado.
                                                                c := 7630058
> d:= M; #Operación de descifrado: c^d mod (n).
                                                                d := 6447
```

Con números más grandes aún obtenidos a partir del software específico Expocrip. Los pasos seguidos serían exactamente los mismos que en el ejemplo anterior

** Esperamos que las capturas de pantalla sean lo suficientemente claras y visibles, según los pasos explicados antes de la primera. Ha sido la única forma de conseguir que cupiera en el texto.

Para poner en práctica los ejemplos desarrollados con el método de encriptación RSA, optamos por cifrar como ejemplo el nombre con el que hemos concursado "Primos de Fermat". Las dos primeras letras las hemos codificado y descodificado manualmente, y para el resto, hemos optado por emplear un software encontrado por internet "ExpoCrip", para aliviar tan tediosa tarea. Aparece la conversión al sistema hexadecimal, que en las dos primeras letras hemos realizado manualmente, puesto que el mencionado software, proporciona la codificación en este sistema.
Para cifrar estas letras hemos usado como parámetros:

$p = 277 \quad q = 307 \quad e = 311$

$n = p * q = 277 * 307 = 85039$

$\Phi(n) = (p-1) * (q-1) = 276 * 306 = 84456$

$d = inv[e, \Phi(n)] = inv\ (311, 84456) = 80111$

Los parámetros p y q, los hemos sacado de una tabla de números primos del programa "ExpoCrip".
Cifrar P:
P en base 10 = 80, para realizar esta conversión utilizamos otra tabla del programa "ExpoCrip" con la conversión entre la letra y el número en código ANSI.

$80^{311}\ mod\ (85039) = 2037$

Para pasar de decimal a hexadecimal seguimos los siguientes pasos:

2037	16		
5	127	16	
	15	7	
5	F (15 = F)	7	
Unidades	Decenas	Centenas	

El número en hexadecimal es 7F5.

Cifrar r:
r en base 10 = 114

$114^{311}\ mod\ (85039) = 30815$

Volvemos a pasar este número a hexadecimal:

30815	16			
15	1925	16		
	5	120	16	
		8	7	
F (15 = F)	5	8	7	
Unidades	Decenas	Centenas	Millares	

El número resultante en hexadecimal es: 785F
30815 / 16 = 1925 de cociente y 15 de resto (este 15 ocupa el lugar de las unidades y se convierte en una F), 1925 / 16 = 120 de cociente y 5 de resto (el 5 ocupa el lugar de las decenas), 120 / 16 = 7 de cociente y 8 de resto (que ocupa el lugar de las centenas, y el 7 ocupa el lugar de los millares). Por lo tanto el numero resultante n hexadecimal es 785F.
A continuación una tabla con las letras de "Primos de Fermat" codificadas:

	P	r	i	m	o	s		d	e		F	e	r	m	a	t
$Cifrado_{16}$	7F5	785F	7AB8	DE04	441A	14382	12B98	DEF0	A0C2	12B98	7C85	A0C2	785F	DE04	53D	E71
$Cifrado_{10}$	2037	30815	31416	56836	17434	82818	76696	57072	41154	76696	31877	41154	30815	56836	1341	5916

Como quedaría cada cifrado:
16 = 7F5 78F5 7AB8 DE04 441A 14382 12B98 DEF0 A0C2 12B98 7C85 A0C2 78F5 DE04 53D E719
10 = 2037 30815 31416 56836 17434 82818 76696 57072 41154 76696 31877 41154 30815 56836 1341 59161

Aquí creemos conveniente dar una explicación acerca de qué es el sistema hexadecimal, que utilizan los distintos softwares criptográficos que hemos utilizado.

Sistema hexadecimal

El sistema hexadecimal, a veces abreviado como hex, es el sistema de numeración posicional de base 16 empleando por tanto 16 símbolos. Estos símbolos son: {0, 1, 2,3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F}. Siendo A = 10, B = 11, C = 12, D = 13, E = 14, F = 15.
Conversión:

- De decimal a hexadecimal:
Se coge el numero y se divide por 16, el resto es el numero de las unidades, y el cociente se vuelve a dividir por 16, siendo el resto el numero de decenas, y asi sucesivamente hasta que el cociente sea menor que 15 (incluido).
Ejemplo:
650: 650 / 16 = 40 (cociente), 10 (resto, este número es el número de las unidades). 40 / 16 = 2 (cociente), 8 (resto, ocupa el lugar de las decenas). 2, al ser menor que 15 se queda como 2 en el lugar de las centenas. De tal forma el número en hexadecimal se quedaría: **28A**
2481: 2481 / 16 = 155 (cociente), 1 (resto). 155 / 16 = 9 (cociente), 11 (resto). Resultado: **9B1**
De hexadecimal a decimal:
$XYZ = X * 16^2 + Y * 16^1 + Z * 16^0$
XYZ son distintos valores entre 0 y F, y el numero al cual está elevado 16 es el lugar en que ocupa XYZ, es decir, se eleva a 0 si está en el lugar de las unidades, a 1 si está en el lugar de las decenas, a 2 si está en el lugar de las centenas, y así sucesivamente.
12: Y = 1, Z = 2; $1 * 16^1 + 2 * 16^0 = 16 + 2 = \mathbf{18}$
135: X = 1, y = 3, Z = 5; $1 * 16^2 + 3 * 16^1 + 5 * 16^0 = 256 + 48 + 5 = \mathbf{309}$
4E: Y = 4, Z = E (14); $4 * 16^1 + 14 * 16^0 = 64 + 14 = \mathbf{78}$

11.- CONCLUSIONES:

Y hasta aquí todo nuestro trabajo.
Aunque en un principio no teníamos ni idea de lo que íbamos a conseguir, hemos adquirido muchos conocimientos acerca de los que, en principio no habíamos oído ni hablar.
Nos hemos introducido en unas matemáticas que esperamos nos sirvan en el futuro y, para qué negarlo, nos gustaba eso de saber cosas que a nuestros compañeros les sonaban a chino.
También nos ha servido para trabajar en equipo de una forma que nunca antes habíamos hecho, aprender a valorar y respetar las opiniones e ideas de los demás.

También hemos aprendido a crecernos ante las dificultades y a sacar tiempo de donde no lo había: contenidos en un principio desconocidos, los consabidos exámenes del colegio, que hacían que nos agobiáramos y discutiéramos entre nosotros, aunque sin llegar nunca la sangre al río. Hemos visitado muchas veces la biblioteca en la búsqueda de libros que nos pudieran ayudar. Descubrimos la falta de fondos de libros específicos en las bibliotecas normales, lo que nos llevó a la biblioteca de la Facultad de Matemáticas y aunque aquí si vimos cosas que nos podrían valer, la mayoría o eran muy avanzados o menos claros que lo que ya teníamos por lo que al final casi siempre acabábamos usando el primero que nos aconsejaron. Aquí tuvimos que vencer la resistencia que teníamos a utilizar bibliografía en inglés. Nos parecía que si bastante poco entendíamos ya, en inglés sería horrible, pero comprobamos que con el tiempo nos acostumbramos y para lo que nosotros necesitábamos entendíamos lo suficiente. Al fin y al cabo si nos decían que era el mejor, tal vez tuvieran razón ¿no?
Aprendimos a trabajar con software científico adecuado como Maple que nos permitió operar con números más grandes y conseguir ejemplos más complejos que los que podíamos hacer a mano con números sencillos.
Para cifrar y descifrar, comprobar si lo que hacíamos funcionaba, utilizamos un software especial que encontramos en una de las innumerables búsquedas en Internet también sufrimos para aprender a trabajar con él, aunque dicho sea de paso no conseguimos sacarle todas las utilidades que probablemente tiene.

12.- BIBLIOGRAFÍA

Páginas y referencias de internet

- http://www.criptored.upm.es/guiateoria/gt_m001a.htm -
- http://www.criptored.upm.es/software/sw_m001k.htm
- http://thales.cica.es/rd/Recursos/rd97/UnidadesDidacticas/16-2-o-primos.html
- http://www.telefonica.net/web2/lasmatematicasdemario/Aritmetica/Numeros/Divisibilidad/Primos/PolGenPri.htm
- http://it.aut.uah.es/enrique/docencia/ii/seguridad/documentos/t4-0506.pdf
- http://www.algebra.com/algebra/homework/equations/Diophantine_equation
- http://huitoto.udea.edu.co/SistemasDiscretos/contenido/alg_euclides.html
- http://mathworld.wolfram.com/DiophantineEquation.html
- http://www.revistasic.com/revista40/agorarevista_40.htm
- Libro Electrónico De Seguridad Informática Y Criptografía
  (Autor: Jorge Ramió Aguirre,Universidad Politécnica de Madrid)
- Enciclopedia.us.es/index.php/Números_coprimos
  boards5.melodysoft.com/app?ID=canalingenio&msg=225
- www.mundocripto.com

Libros

- David M. Burton - Elementary Number Theory. Editorial Allyn and Bacon
- Pino Caballero Gil – Introducción a la Criptografía. Editorial Ra-Ma

13.- AGRADECIMIENTOS

En nombre de LOS PRIMOS DE FERMAT, queremos dar las gracias a todos los que nos han ayudado en esas interminables horas de búsqueda de información: Jorge Ramió Aguirre, Alfonso Muñoz y Silvia Teresita Acuña. Nos recomendaron una serie de páginas y libros esclarecedores que nos sirvieron de mucho.
También queremos agradecer a nuestros profesores la paciencia que han tenido con nosotros cuando nos entraban nuestros tradicionales ataques de histeria cuando no sabíamos hacer nada y nos agobiábamos o cuando nos querían reunir para trabajar y teníamos un control de los miles que habitualmente hacemos. Ha habido veces en que nos han tenido que empujar o animar para que siguiéramos. En definitiva, al final no se han portado tan mal con nosotros.