PRÁCTICA 1

PRIMALIDAD

En esta práctica vamos a trabajar un concepto muy importante en la criptografía actual: la primalidad. Dado un número impar, intentaremos decidir si es primo o no.

En el desarrollo de esta práctica, y otras posteriores, se van a utilizar números de gran tamaño, por lo que es necesario optar por un lenguaje de programación que admita dichos cálculos. Aunque también es posible utilizar otros lenguajes que puedan ser adaptados, mediante el uso de bibliotecas específicas, para la realización de estos cálculos.

Va a ser necesario un algoritmo que calcule potencias modulares de forma eficiente. Este algoritmo lo vimos en la primera sesión de prácticas.

Una vez que ya tengamos este algoritmo implementado, realizamos algunos cálculos:

2^{28} mód $29 = 1$	5^{28} mód $29 = 1$	11^{28} mód $29 = 1$
2^{30} mód $31 = 1$	5^{30} mód $31 = 1$	11^{30} mód $31 = 1$
2^{32} mód $33 = 4$	5^{32} mód $33 = 25$	11^{32} mód $33 = 22$
2^{34} mód $35 = 9$	5^{34} mód $35 = 30$	11^{34} mód $35 = 11$
2^{36} mód $37 = 1$	5^{36} mód $37 = 1$	11^{36} mód $37 = 1$

En todos los casos, hemos elegido un número impar n, y hemos calculado 2^{n-1} mód n, 5^{n-1} mód n y 11^{n-1} mód n.

Para los casos en que n es primo ($n = 29, 31, 37$) el resultado ha sido siempre 1, mientras que si n no es primo ($n = 33, 35$) el resultado es distinto de 1.

Esto es consecuencia del teorema (pequeño) de Fermat, que afirma que si p es un número primo y $1 \leq a \leq p - 1$ entonces $a^{p-1} \equiv 1$ mód p.

Para ver si un número n es o no primo, podemos, en lugar de buscar sus divisores, intentar calcular a^{n-1} mód n para algún número a. Si nos da distinto de 1, sabemos que el número n no es primo.

Por ejemplo, tomamos $n = 23973$. Puesto que 23^{23972} mód $23793 = 4900 \neq 1$ deducimos que 23793 no es primo. También podíamos haber visto que 2^{23972} mód $23973 = 3595 \neq 1$.

Si al calcular a^{n-1} mód n nos sale distinto de 1 podemos concluir que el número n no es primo. Pero, ¿y si a^{n-1} mód $n = 1$?

En tal caso, no sabemos si el número es o no primo. En los casos que hemos estudiado más arriba, hemos visto que cuando a^{n-1} mód $n = 1$ entonces n era un número primo.

Pero, por ejemplo, se tiene que 3^{90} mód $91 = 1$ (compruébalo), y sin embargo 91 no es primo ($91 = 7 \cdot 13$). O también se tiene que 2^{340} mód $341 = 1$ y 341 tampoco es primo (pues $341 = 11 \cdot 13$).

En el caso del 91 se podía haber llegado a la conclusión de que no es primo si hubiéramos calculado 2^{90} mód 91, pues el resultado de esta operación es 64. Y para el 341, si hubiéramos calculado 3^{340} mód 341 también habríamos visto que 341 no es primo.

A la vista de estos ejemplos, una opción para saber si n es o no primo es elegir varios números a entre 2 y $n - 2$ y para ellos calcular a^{n-1} mód n. Si en algún caso el resultado es distinto de 1, el número n sabemos que no es primo. Si en todos los casos el resultado es 1, el número será probablemente un número primo.

Pero incluso en estos casos, esta prueba podría fallar. Toma $n = 2199733160880$ (que no es un número primo), elige al azar unos cuantos valores de a (por ejemplo, elige 20 valores distintos), y calcula para cada uno de ellos a^{n-1} mód n.

Es bastante probable que en todos los casos el resultado sea 1, luego, de usar este criterio, llegaríamos a una conclusión errónea.

Para solucionar esto vamos a fijarnos en otro hecho. En primer lugar, dado un número impar n (si n es par sabemos que no es primo), vamos a encontrar todos los números x, con $1 \leq x \leq n - 1$, tales que x^2 mód $n = 1$.

n	$x : x^2$ mód $n = 1$
29	1, 28
31	1, 30
33	1, 10, 23, 32
35	1, 6, 29, 34
37	1, 36

Y vemos que la ecuación $x^2 - 1$ tiene siempre las soluciones 1 y $n - 1$ módulo n. Si n es primo, estas son las dos únicas solucinones, pero cuando n no es primo, la ecuación tiene más soluciones.

De hecho, se tiene que si n es un número impar que es divisible por r primos distintos, entonces la ecuación $x^2 - 1 = 0$ tiene 2^r soluciones módulo n (en el caso de que n sea una potencia de un primo, por ejemplo, $n = 27 = 3^3$, la ecuación $x^2 - 1 = 0$ tendría sólo dos soluciones).

Con esto en mente, volvemos a un caso que hemos estudiado previamente. Hemos visto que 3^{90} mód $91 = 1$, y eso nos hacía llegar a la conclusión de que 91 podría ser primo.

Sin embargo, $3^{90} = (3^{45})^2$, y 3^{45} mód $91 = 27$. Hemos encontrado un elemento, 27, que es solución de la ecuación $x^2 - 1 = 0$. Al haber encontrado una solución de esta ecuación que no es ni 1 ni $91 - 1$, podemos asegurar que el número 91 no es primo.

Vamos a tomar $n = 1729$, y elegimos distintos valores de a. Por ejemplo, $a = 2$, $a = 3$, $a = 5$ y $a = 10$. Comprobamos que en todos los casos se tiene que a^{1728} mód $n = 1$.

Sin embargo, como $1728 = 2^6 \cdot 27$ vamos a realizar el cálculo de a^{1728} en varias etapas.

En primer lugar calculamos a^{27} mód 1729. Luego calculamos el cuadrado de este número módulo 1729, con lo que tendremos a^{54} mód 1729. Volvemos a calcular el cuadrado, y así tendremos a^{108} mód 1729. Y de esta forma, después de elevar 6 veces al cuadrado tendremos a^{1728} mód 1729.

Los cálculos los ordenamos en la siguiente tabla:

a	a^{27}	a^{54}	a^{108}	a^{216}	a^{432}	a^{864}	a^{1728}
2	645	1065	1	1	1	1	1
3	664	1	1	1	1	1	1
5	1217	1065	1	1	1	1	1
10	1728	1	1	1	1	1	1

Y vemos que en el cálculo de 2^{1728}, que vale 1, hemos pasado por $2^{27} = 645$, $2^{54} = 645^2 = 1065$ y $2^{108} = 1065^2 = 1$. Y con esto, hemos encontrado una solución de la ecuación $x^2 - 1 = 0$ que no es ni $x = 1$ ni $x = 1728$. Estos cálculos nos dirían entonces que 1729 no es primo.

Entonces, aunque $2^{1728} = 1$, lo que nos haría pensar que 1729 podría ser primo, al realizar el cálculo en estas etapas vemos que 1729 no es primo.

Algo similar nos ocurre con $a = 3$ (aquí vemos que $664^2 = 1$) y con $a = 5$. Sin embargo, con $a = 10$, estos cálculos nos dicen que el número 1729 podría ser primo, pues la solución que hemos encontrado de $x^2 - 1 = 0$ es $x = 1728$.

Si tomáramos $a = 92$ veríamos que $a^{27} = 1$, por lo que sólo con esa información el número 1729 podría ser primo.

Por último, si tomáramos $a = 13$ en este caso tenemos que $a^{1728} = 533 \neq 1$, luego nos dice que el número 1729 no es primo.

En resumen, para estudiar si un número impar n es primo hacemos lo siguiente:

- Descomponemos $n - 1$ como una potencia de 2 por un número impar. Es decir, $n - 1 = 2^u s$, con s impar.
- Elegimos un número a tal que $2 \leq a \leq n - 2$. Este número es lo que llamaremos un testigo.
- Calcularemos (módulo n) las siguientes potencias:

$$a^s, \qquad a^{2s}, \qquad a^{2^2 s}, \qquad \cdots \qquad a^{2^{u-1} s}.$$

  Notemos que cada elemento de esta lista (salvo el primero) es el cuadrado del elemento anterior.

  Y ahora:

  - Si todas las potencias valen 1 (o lo que es lo mismo, si $a^s = 1$), el número n podría ser primo.
  - Si en esa lista encontramos una potencia que vale $n - 1$ el número n podría ser primo.

- Si en la lista encontramos una potencia que vale 1 (sin que la anterior valga $n-1$), el número no es primo (ya que el elemento anterior de la lista será una solución de $x^2 - 1 = 0$).
- Si llegamos al final de la lista y no hay ningún 1 ni ningún $n-1$ el número n no es primo (en este caso, podría ocurrir que $a^{2^u s} = a^{n-1} = 1$, en cuyo caso, el cuadrado del último elemento de la lista sería una solución de $x^2 - 1 = 0$ o $a^{2^u s} = a^{n-1} \neq 1$. En ambos casos, el número n no puede ser primo).

Dado un número impar n y un testigo a ($2 \leq a \leq n-2$), el test de Miller-Rabin consiste en realizar los cálculos antes descritos. Diremos que el test falla si nos dice que un número n puede ser primo y sin embargo no lo es.

Por ejemplo, el test de Miller-Rabin falla para el número 1729 y el testigo $a = 10$. En tal caso diremos que $a = 10$ es un testigo falso.

Para minimizar la probabilidad de error, lo que se hace es, dado un número n, elegir varios testigos. Si para algún testigo el test nos dice que el número n no es primo, tendremos seguridad que n es compuesto. Si para todos los testigos nos dice que n podría ser primo, admitiremos n como un número primo.

Si n es un número impar y compuesto, la probabilidad de encontrar un testigo falso es menor que $\frac{1}{4}$. En tal caso, si n es un número impar y elegimos m testigos tales que todos ellos nos dicen que n podría ser primo, la probabildad de error si tomamos n como número primo es menor que $\frac{1}{4^m}$ (por ejemplo, para $m = 20$ tenemos una probabilidad de error menor que 1 entre un billón).

En esta práctica vamos a estudiar el test de Miller-Rabin, y vamos a analizar cuál es la probabilidad de que el test falle en algunos casos concretos.

El trabajo que hay que realizar es el siguiente.

1. Dado un número impar n tenemos que decidir si n es primo o no (con un pequeño margen de error). Para esto, usaremos el test de Miller-Rabin, y tenemos que decidir cuántos testigos elegimos para que la probabilidad de error sea pequeña. Si se quiere, antes de pasar el test de Miller-Rabin podemos probar a dividir el número n por los primeros números primos. Si en algún caso la división es exacta, ya tendríamos que el número no es primo.

2. Tenemos que implementar una versión del test de Miller-Rabin que nos permita hacer pruebas. Hay que poder decidir para un número n compuesto e impar, si un número a es un falso testigo o no lo es.

3. Hay que hacer una función que, dado un número n, calcule el siguiente número primo (mejor dicho, el siguiente probable primo).

4. Hay que hacer una función que, dado un número n, calcule el siguiente número primo fuerte. Un número p es un primo fuerte si tanto él como $\frac{p-1}{2}$ son números primos.

5. Dado un número n hay que calcular un número primo (fuerte) con n bits (es decir, un primo p tal que $2^{n-1} \leq p < 2^n$).

6. Elegimos un número de la siguiente lista:

$$6601, \quad 8911, \quad 10585, \quad 15841, \quad 29341.$$

   Para el número elegido tenemos que encontrar todos los falsos testigos.

7. Elegimos dos números compuestos grandes (uno que sea producto de varios números primos pequeños y otro que sea producto de dos números primos grandes. Para elegir estos primos usaremos alguna de las funciones que hemos implementado en los aparatados 3, 4 ó 5). Para estos dos primos, elegiremos al azar doscientos testigos y veremos cuáles son falsos.

8. Repetimos el apartado anterior pero con los números $n_1 = 3215031751$ y $n_2 = 2199733160881$.

De la práctica hay que entregar:

El código fuente.

Una memoria que incluya instrucciones para su ejecución en Linux y los falsos testigos que se han encontrado en los distintos apartados.

Todo esto se subirá a SWAD en la sección de *Archivos - Trabajos* o se enviará por correo electrónico.
Una vez entregada, hay que realizar una defensa de la misma.
La fecha límite de entrega es el día 17 de marzo.