GUIONES DE CRIPTOGRAFÍA Y COMPUTACIÓN

ARITMÉTICA MODULAR

Todos los cálculos son con aritmética de múltiple precisión, por lo que deberíais elegir un lenguaje que la soporte.

Ejercicio 1. Implementa el algoritmo extendido de Euclides para el cálculo del máximo común divisor: dados dos enteros a y b, encuentra $u, v \in \mathbb{Z}$ tales que $au + bv$ es el máximo común divisor de a y b (véase por ejemplo [1, Algorithm 2.107]).

Podéis comprobar los resultados comparando con `GcdRepresentation` de `GAP` ([2]).

Ejercicio 2. Usando el ejercicio anterior, escribe una función que calcule a^{-1} mód b para cualesquiera a, b enteros que sean primos relativos.

Ejercicio 3. Escribe una función que calcule a^b mód n para cualesquiera a, b y n enteros positivos. La implementación debería tener en cuenta la representación binaria de b ([1, Algorithm 2.143]).

Se pueden hacer pruebas con la función en `GAP` `PowerModInt`.

Ejercicio 4. Dado un entero p, escribe una función para determinar si p es (probablemente) primo usando el método de Miller-Rabin ([3, C.9.5]).

`IsPrime` determina si el argumento es primo en `GAP`.

Ejercicio 5. Implementa el algoritmo paso enano-paso gigante para el cálculo de algoritmos discretos en $\mathbb{Z}_p$ (véase por ejemplo [1, 3.56]).

Ejercicio 6. Sea $n = pq$, con p y q enteros primos positivos.

- Escribe una función que, dado un entero a y un primo p con $\left(\frac{a}{p}\right) = 1$, devuelve r tal que $r^2 \equiv a$ mód p ([3, §2.3.4]; primero te hará falta implementar el símbolo de Jacobi [1, 2.149]).
- Sea a un entero que es residuo cuadrático módulo p y q. Usa el teorema chino de los restos para calcular todas las raíces cuadradas de a mod n a partir de las raíces cuadradas de a módulo p y q.

Ejercicio 7.

- Implementa el Método de Fermat para factorización de enteros.
- Implementa el algoritmo de factorización ρ de Pollard ([1, 3.9]).

Ejercicio 8. Compara los tiempos de ejecución de tus implementaciones con las de tus compañeros y con las primitivas de algunos paquetes de cálculo simbólico como (`GAP`, MATHEMATICA, `maxima`, ...).

REFERENCIAS

[1] A. Menezes, P. van Oorschot, and S. Vanstone, Handbook of Applied Cryptography, CRC Press, 1996.
[2] GAP system.
[3] S. Goldwasser, M. Bellare, Lecture Notes on Cryptography

FIGURA 1. Lista de primos para hacer pruebas

46381, 768479, 9476407, 36780481, 562390847, 1894083629,
65398261921, 364879542899, 8590365927553, 28564333765949, 123456789101119