UVG-MM2015:
Matemática Discreta
Instructor: Héctor Villafuerte

Proyecto de RSA

5 de Noviembre, 2009

Nombre (y Carnet): ____________________

1. RSA es un popular criptosistema que utiliza conceptos básicos de teoría de números que se presentan a continuación:

   (a) Un entero positivo $n > 1$ es *primo* si es divisible sólo por 1 y n.

   (b) Sean a, b enteros, no ambos cero. Entonces decimos que el *máximo común divisor* ("greatest common divisor") de a, b es el mayor entero positivo que es un factor de ambos a, b. Usualmente se denota $\gcd(a, b)$.

```
sage: gcd(18,27)
9
sage: gcd(11,17)
1
```

   (c) Factorice 18 en primos. Factorice 27 en primos. ¿Qué puede decir ahora de $\gcd(18, 27)$?

   (d) Factorice 11 en primos. Factorice 17 en primos. ¿Qué puede decir ahora de $\gcd(11, 17)$?

   (e) Si $\gcd(a, b) = 1$ decimos que a, b son *coprimos* (o *primos relativos*).

```
sage: gcd(32,45)
1
```

   (f) Factorice 32 en primos. Factorice 45 en primos. ¿Qué puede decir ahora de $\gcd(32, 45)$?

   (g) Al trabajar con división de enteros el residuo de la operación es una buena caracterización. Esta es la idea detrás de *aritmética modular* y *congruencia módulo* n. Sean $a, b, n \in \mathbb{Z}$ con $n \neq 0$. Si a, b tienen el mismo residuo entero al dividir por n, entonces decimos que a, b son *congruentes módulo* n y denotamos esta relación por $a \equiv b(\mod n)$. Otra forma de expresar la división entera a/n es $a = nq + r$ donde $n, q, r \in \mathbb{Z}$.

   (h) Considere la división entera de $8/5$ y $23/5$. ¿Son $8 \equiv 23(\mod 5)$?

```
sage: mod(8,5)
3
sage: mod(23,5)
3
```

   (i) Dado un entero m, queremos encontrar los enteros n, $1 \leq n \leq m$, tal que $\gcd(n, m) = 1$. En palabras, los coprimos n menores a m.

```
sage: m = 18
sage: [n for n in range(1,m) if gcd(n,m)==1]
[1, 5, 7, 11, 13, 17]
sage: m = 11
sage: [n for n in range(1,m) if gcd(n,m)==1]
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
```

   Si sólo queremos el número de coprimos menores de m podemos usar la *función phi de Euler*, $\phi(m)$.

```
sage: euler_phi(18)
6
sage: euler_phi(11)
10
```

Esta función también es llamada “Euler’s totient”.

2. El algoritmo RSA para encriptar y decriptar básicamente se aprovecha de la gran diferencia en el costo computacional que existe entre *multiplicar enteros* (barato) y *factorizar un entero en primos* (caro). Los pasos del algoritmo son los siguientes:
   (a) Tome dos primos p_1, p_2 y sea $n = p_1 \cdot p_2$.
   (b) Sea $e \in \mathbb{N}$ tal que $\gcd(e, \phi(n)) = 1$.
   (c) Calcule un valor para $d \in \mathbb{Z}$, tal que $d \cdot e \equiv 1 (\mod \phi(n))$.
   (d) La *llave pública* es entonces la pareja (n, e). La *llave privada* es entonces la tripleta (p_1, p_2, d).
   (e) Para un entero $m < n$, encripte usando $c \equiv m^e (\mod n)$.
   (f) Decripte c usando $m \equiv c^d (\mod n)$.
3. Encripte la palabra “PUCHICA” usando el algoritmo RSA. Detalle sus pasos.