CSC110 Lecture 19: Public-Key Cryptography and the RSA Cryptosystem

Print this handout

For your reference, here is one key definition and the two main theorems about modular exponentiation that we'll use today.

(*Fermat's Little Theorem*) Let $p, a \in \mathbb{Z}$ and assume p is prime and that $p \nmid a$. Then $a^{p-1} \equiv 1 \pmod{p}$.

We define the function $\varphi : \mathbb{Z}^+ \to \mathbb{N}$, called the **Euler totient function** (or **Euler phi function**), as follows:

$$\varphi(n) = \big|\{a \mid a \in \{1, \ldots, n-1\},\ \text{and } \gcd(a, n) = 1\}\big|.$$

We have the following formulas for special cases of $\varphi(n)$:

- For all primes $p \in \mathbb{Z}^+$, $\varphi(p) = p - 1$.
- For all *distinct* primes $p, q \in \mathbb{Z}^+$, $\varphi(pq) = (p-1)(q-1)$.

(*Euler's Theorem*). For all $a \in \mathbb{Z}$ and $n \in \mathbb{Z}^+$, if $\gcd(a, n) = 1$ then $a^{\varphi(n)} \equiv 1 \pmod{n}$.

Also recall: $\forall a, b, c, d, n \in \mathbb{Z},\ n \neq 0 \land a \equiv b \pmod{n} \land c \equiv d \pmod{n} \Rightarrow ac \equiv bd \pmod{n}$

Exercise 1: Reviewing modular exponentiation

1. Let $a, p \in \mathbb{Z}$ and assume that p is prime that $\gcd(a, p) = 1$. Using Fermat's Little Theorem, simplify each of the following expressions modulo p by reducing it to 1 or an expression of the form a^e, where the exponent e is positive and as small as possible. We've done the first one for you.

Power of a	**Simplified expression modulo p**
a^{p-1}	1
$a^p = a^{p-1} \cdot a$	Since $a^{p-1} \equiv 1 \pmod{p}$ and $a \equiv a \pmod{p}$

Fermat Little Thm

$(a^{p-1})a \equiv a \pmod{p}$

p

or $a \equiv a \pmod p$

Power of a	Simplified expression modulo p
$a^{2p-2} = a^{2(p-1)} = a^{p-1} \cdot a^{p-1}$	$a^{2p-2} \equiv 1 \pmod p$
$\boxed{a^{2p}} = a^{p-1} \cdot a^{p-1} \cdot a^2$	$a^{2p} \equiv a^2 \pmod p$
$a^{p^2-1} = (a^{p-1})^{p+1}$	$a^{p-1} \equiv 1 \pmod p$
$a^{p^2} = a^{p^2-1} \cdot a$	$a^{p^2} \equiv a \pmod p$

$p^2 - 1 = (p-1)(p+1)$

$a^{2p} = a^p \cdot a^p = (a^{p-1} \cdot a)(a^{p-1} \cdot a) = (a^{p-1})^2 \cdot a$

2 Let $p = 23$ and $q = 5$. $\quad p \cdot q = 115$

a. What is $\varphi(pq)$?

Since 23 is prime and 5 is prime we know

$\varphi(pq) = \varphi(23 \cdot 5) = (23-1) \cdot (5-1) = 22 \cdot 4 = 88$

b. Using Euler's Theorem, calculate each of the following remainders (modulo $pq = 115$). We have dor the first row for you (note that $(p-1)(q-1) = 88$—keep this number in mind).

Power of 2	Remainder modulo $pq = 115$
2^{88}	1
$2^{89} = 2^{88} \cdot 2^1$	$2^{89} \equiv 2^1 \bmod 115$
$2^{176} = 2^{88} \cdot 2^{88}$	$2^{176} \equiv 1 \bmod 115$
$2^{180} = 2^{176} \cdot 2^4$	$2^{180} \equiv 2^4 \bmod 115$ or $2^{180} \equiv 16 \bmod 115$
$2^{880} = (2^{88})^{10}$	$2^{880} \equiv 1 \bmod 115$
$2^{8801} = (2^{88})^{100} \cdot 2$	$2^{8801} \equiv 2 \bmod 115$

Exercise 2: Reviewing the RSA Cryptosystem

1. The following parts get you to manually trace through the steps of the RSA cryptosystem. The calculations themselves are pretty straightforward, we just want you to review the algorithm and practice all of the steps!

   a. Suppose we start with the primes $p = 23$ and $q = 5$. What are n and $\varphi(n)$?

$n = 23 \cdot 5 = 115$

$\varphi(n) = (23-1)(5-1)$

$= 22 \cdot 4$
$= 88$

b. Suppose $e = 3$. Find the corresponding value for d such that $ed \equiv 1 \pmod{\varphi(n)}$. (You can just use trial and error here, or the `modular_inverse` function from an earlier lecture!)

$147 \% 88 = 59$

$147 = 1 \cdot 88 + 59$

$d = 59$

$3 \cdot 59 = 177$
$= 2 \cdot 88 + 1$

c. What are the RSA private and public keys for these choices of p, q, and e?

public $(n, e) = (115, 3)$

private $(p, q, d) = (23, 5, 59)$

d. Suppose you want to encrypt the number 77 using the public key. What is the resulting "ciphertext" (the encrypted number)? You can use Python as a calculator to answer this.

$C = \text{pow}(77, 3, 115)$
$= 98$

147
$= 1 \cdot$

e. Verify that if you decrypt this ciphertext with the private key, you get back the original number 77.

$m' = \text{pow}(C, 59, 115)$
$= 77!$

2. The following are some conceptual questions about the RSA algorithm to check your understanding this algorithm.

a. Why does the key generation phase require that $\gcd(e, \varphi(n)) = 1$?

Because we want to be able to find d, the modular inverse of e mod $\varphi(n)$. We can ensure that we can find d by having $\gcd(e, \varphi(n)) = 1$.

b. We know that picking $e = 1$ satisfies $\gcd(e, \varphi(n)) = 1$. Yet why is $e = 1$ not a good choice?

It makes it easy to guess a correct d.

$d = 1$ works!

large n makes $\varphi(n)$ large and $e \neq 1$

makes guessing d hard.

c. When we discussed encrypting a message, we said that the message had to be in the range $\{1, 2, \ldots, n-1\}$ (where the n is from the public key). Why do we not allow numbers larger tha $n-1$ to be encrypted?

Since $2 \equiv 2+n \mod n$

$$2^e \equiv (2+n)^e \mod n$$

when decrypting

$$(2^e)^d \equiv ((2+n)^e)^d \mod n$$

is the original message 2 or 2+n ?!