# Announcement

- Homework 1 was due 30 seconds ago…
- Project 1 is due at 7PM on Tuesday, 3/8.
- Homework 2 will start on Tuesday, 3/1.
- TA: Anu Aggarwal <anuagg0102@gmail.com>
- Contact me by tomorrow if you want to give the senior oral presentation in this class

# Last Time

- DES: 64-bit plaintext, 64-bit ciphertext, 64-bit key
- ECB vs. CBC
- Hash Functions: one-way, collision resistant

# Message Authentication Code

- Authenticate the integrity of messages
  - Given hash function $h()$, key $k$, and message $m$
    $MAC(k, m) = h(m|k)$
  - Send both message $m$ and the message authentication code $MAC(k, m)$ to the receiver
  - The receiver computes $h(m|k)$ using the received message and compares the result with the received $MAC(k, m)$
- Q: Why does $MAC(k, m)$ provide integrity?
  - Cannot generate $MAC(k, m)$ without knowing the key $k$
- Can we use $h(m)$ instead?

# Public Key Cryptography

- Each individual has two keys: a public key $k^+$ known to everyone, a private key $k^-$ kept secret to the owner
  - $D(E(m, k^+), k^-) = m; D(E(m, k^-), k^+) = m$
- Everyone can use the receiver's public key to encrypt a message, and only the receiver can use his private key to decrypt it
  - $E(m, k^+) = c$ *and* $D(c, k^-) = m$
- Digital Signature
  - $E(m, k^-) = c$ *and* $D(c, k^+) = m$
- Also known as asymmetric key cryptography

# Modular Arithmetic

- Use non-negative integers less than some positive integer *n*, perform arithmetic operations, and then replace the result with the remainder when divided by *n*
- Modular Addition
  - Example:

    $6 + 9 \ mod \ 10 = 5 \ mod \ 10$

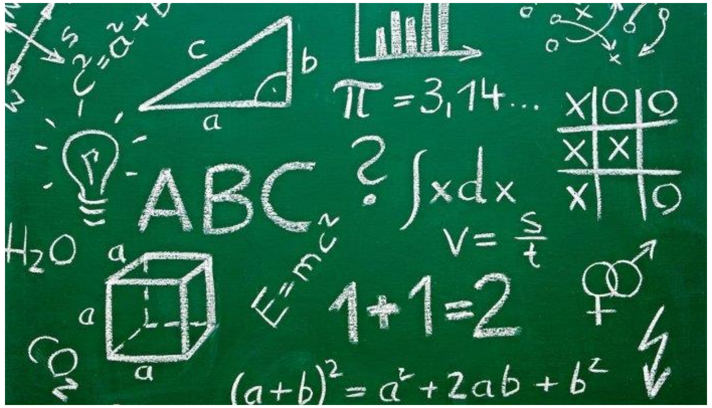
    $3 + 7 \ mod \ 10 = 0 \ mod \ 10$
- Modular Multiplication
- Modular Exponentiation

# Modular Multiplication

- Example: $3 \times 7 = 1 \ mod \ 10$; $5 \times 2 = 0 \ mod \ 10$
- Multiplicative Inverse
  - If $xy = 1$ mod $n$, then $x$ and $y$ are each other's multiplicative inverse mod $n$.
  - Example: 3 is the multiplicative inverse of 7 modular 10
  - A number $x$ has multiplicative inverse mod $n$ if and only if $x$ is relatively prime to $n$
- Totient Function $\phi(n)$
  - The number of numbers that are relatively prime to $n$
  - If $n=pq$ where $p$ and $q$ are prime, $\phi(n)= \phi(pq)=(p-1)(q-1)$
  - Example: $\phi(10)= \phi(2\times5)=(2-1)(5-1)=4$

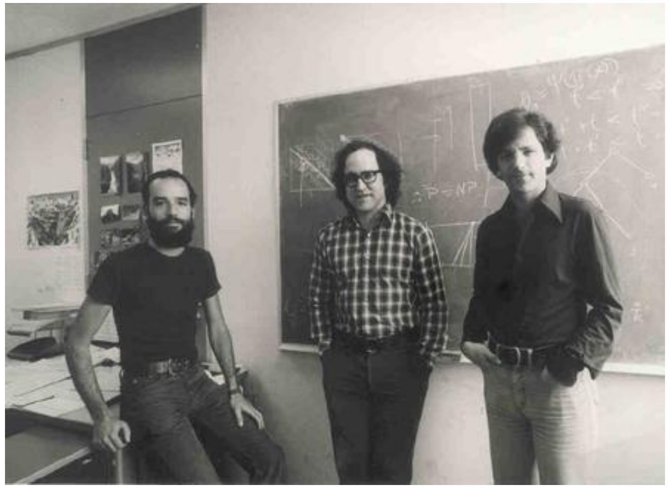
# Modular Exponentiation

- Example: $3^5 = 243 = 3 \ mod\ 10$; $4^6 = 4096 = 6 \ mod\ 10$
- We have $x^y \ mod\ n = x^{(y \ \mathrm{mod}\ \phi(n))} \ mod\ n$
  - $3^5 = 3^{(5 \ mod\ \phi(10))} = 3^{(5 \ \mathrm{mod}\ 4)} = 3 \ mod\ 10$
- If $y = 1 \ mod\ \phi(n)$, then we have $x^y \ mod\ n = x \ mod\ n$



# RSA

- A dominant public key cryptosystem named after Rivest, Shamir, and Adleman
  - The encryption/decryption algorithms are conceptually simple
  - Why it is secure is very deep (number theory)
  - Key length is variable
  - Plaintext block must be smaller than the key length, ciphertext block is the same as the key length



# RSA

- Key Generation
  - STEP1: Pick two large primes $p$ and $q$ (512 bits)
  - STEP2: Calculate $n = pq$
  - STEP3: Choose e such that it is relatively prime to $\phi(n) = (p-1)(q-1)$
  - STEP4: Find d that is the multiplicative inverse of e $mod\ \phi(n)$, i.e., $ed = 1\ mod\ \phi(n)$. (Euclid's Algorithm)
- Example:
  - STEP1: $p = 3$, $q = 11$
  - STEP2: $n = pq = 3 \times 11 = 33$
  - STEP3: $\phi(n) = (p-1)(q-1) = (3-1) \times (11-1) = 20$, choose e = 7
  - STEP4: $3 \times 7 = 1\ mod\ 20$, so $d = 3$

# RSA

- In RSA, public key $k^+$ is <e, $n$> and private key $k^-$ is <$d$, $n$>
- Encryption algorithm: $c = E(k^+, m) = m^e \mod n$
- Decryption algorithm: $m = D(k^-, c) = c^d \mod n$ (why?)
  - Recall $x^y \mod n = x^{(y \mod \phi(n))} \mod n$
- Example:
  - Public key $k^+ = <7, 33>$, Private key $k^- = <3, 33>$
  - Plaintext $m = 4$
  - Encryption: $c = E(k^+, m) = 4^7 \mod 33 = 16384 \mod 33 = 16$
  - Decryption: $m = D(k^-, c) = 16^3 \mod 33 = 4096 \mod 33 = 4$

# Attacks on RSA

- Brute-force attack: try all possible private keys
  - Solution: use a large key space
- Mathematical attack
  - Given $n$ and e, factor $n = pq$. Then, find $\phi(n)$ and $d$.
  - Given $n$ and e, determine $\phi(n)$. Then, find $d$.
  - The second method is equivalent to the first one
  - Fact: factoring large numbers is computationally hard