Quiz 2 - will post sols Tues. Should hand back Wed.

Today: Coding bit-wise operators

$\Rightarrow$ Modern Crypto w/ Computers

Everything is 0/1's

Common ops on bit strings

or $x \lor y$

and $x \land y$

xor $x \oplus y$

What is $x \land x = 0$

If we "encrypt" some plaintext $p$ with a key $x$

$c = \boxed{p \land x}$

$p = c \land x$

$(p \land x) \land x = p$

$= p \land (x \land x)$

$= p \land (0)$
1. Cyclic bit shift

2. Pseudocode

unsigned int x;
array 32 bit

bit shift

\[ x \gg 1 \quad \text{(right shift)} \]

\[ 10111 \gg 1 \Rightarrow 1011 \]

chops off least significant bit (int div by 2)

\[ x \gg k \quad \text{right shift of k bits} \]

\[ x \ll 1 \quad \text{(left shift)} \]

\[ \ll \quad \text{mult by 2} \]

\[ 10111 \ll 1 \Rightarrow 101110 \]

\[ x \ll k \quad \text{(left shift by k bits)} \]

\[ \Rightarrow \text{mult } 2^k. \]

0xf'fffff

31 ones in 32-bit

isolate this bit

isolate this and left shift by 1 bit

if unsigned

bit value

bx \rightarrow xb
80002402
10000000000000000000000000010000000000000100000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
Permutation Example

Input 1 0 1 0 1 1 0 1
       0 1 2 3 4 5 6 7

perm 6 2 0 1 7 5 3 4
     1 1 0 1 1 0 0 6

build backwards // i = index, perm, j = value
for (int i = 31; j = 0; i >= 0; i--, j++) {
    31 - perm[i]; // bitwise location
}

\text{tmp} = \text{Input} \& (1 \ll (31 - \text{perm[i]} - j))

\text{tmp} \gg (31 - \text{perm[i]} - j)

\text{POSITION I want this bit.}