

# Example Class 1

Elementary Number Theory

---

# Outline

- Modulo  $n$
- Binary arithmetic
- Applications of binary arithmetic



# Modulo $n$

- Recall

$$a \equiv b \pmod{n}$$

- If  $a \equiv b \pmod{n}$ , then  $a-b = qn$  and  $a=qn+b$ .
  - We represent integers mod  $n$  as  $\{0,1,\dots,n-1\}$  (thanks to the Euclidean division)
  - We have addition and multiplication modulo  $n$ .
-

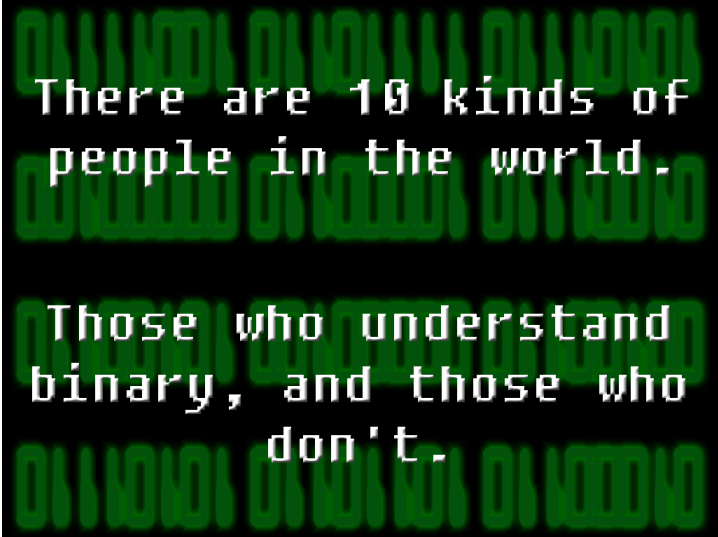
# Integers mod 2

+	0	1
0	0	1
1	1	0

*	0	1
0	0	0
1	0	1

# Counting in Binary

$$\begin{aligned}10010_2 &= 0 \cdot 2^0 + 1 \cdot 2^1 + 0 \cdot 2^2 + 0 \cdot 2^3 + 1 \cdot 2^4 \\&= 18_{10} \\&= 8 \cdot 10^0 + 1 \cdot 10^1\end{aligned}$$



There are 10 kinds of  
people in the world.

Those who understand  
binary, and those who  
don't.

---

# Binary Vectors

- A **vector** is a row (or column) array containing numbers.
  - $(1,0,0,1)$  is a binary row vector (of length 4)
  - One can add two binary vectors component wise (vector addition).
-

# Binary Vectors: Example

- For example  $(1,0,0,1)+(0,0,1,1)=(1,0,1,0)$
  - $S$ =set of binary vectors of length  $n$ ,  $\Delta$ =vector addition. Is  $S$  closed under  $\Delta$ ?
  - Different from counting in binary!
-

# Binary in the Real World

- Storage of data across multiple hard disks
- Data is in binary format.
- Data needs to be stored so as to tolerate disk failures.

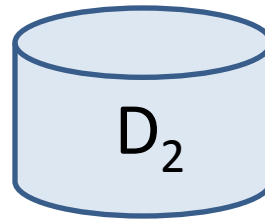
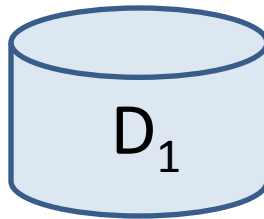
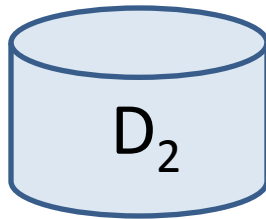
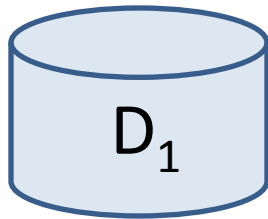




# Example (I)

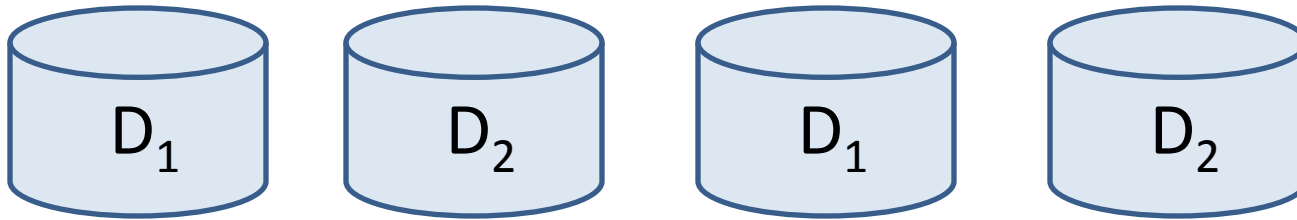
- Suppose you want to store 200GB of (binary) data
- Option 1: buy 4 disks of 100 GB each, store 2 copies of your data.

$$D = (D_1, D_2)$$



# Example (II)

$$D = (D_1, D_2)$$



- Good thing: if one hard disk fails, your data is safe.
  - Bad thing: you paid for 4 hard disks instead of 2.
  - Can we think of a better solution?
-

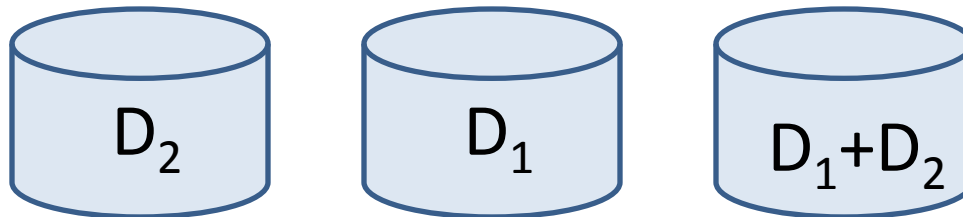
# Example (III)

- Suppose  $a$  and  $b$  are bits, and take  
 $a, b, a+b$

$a$	$b$	$a+b$
0	0	0
0	1	1
1	0	1
1	1	0

- Do the same thing with disks

$$D = (D_1, D_2)$$



# Example (IV): Parity (RAID)

- Binary vector addition:

Drive 1: 01101101  
Drive 2: 11010100

01101101  
**XOR** 11010100  
-----  
**10111001**



Store in Drive 3

You can still loose one disk, but paid for only 3.

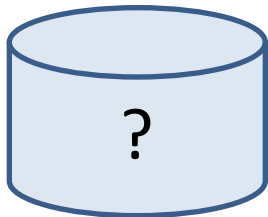
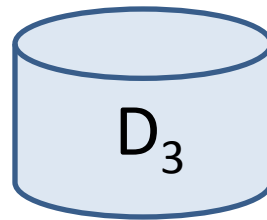
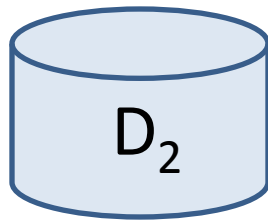
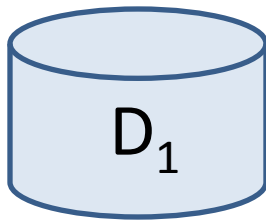
---

# Another Example (I)

- You want to store 150 GB data.
  - Now you are buying storage devices, each of 50GB capacity.
  - This time, even if any arbitrary two devices fail, you still want to recover all your data!
-

# Another Example (II)

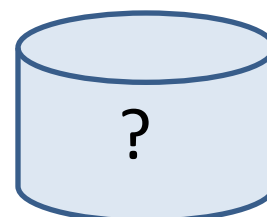
$$D = (D_1, D_2, D_3)$$



$?$

$\dots$

$?$

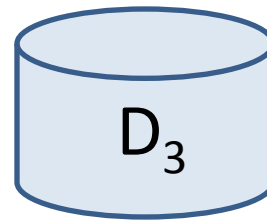
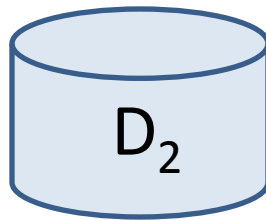
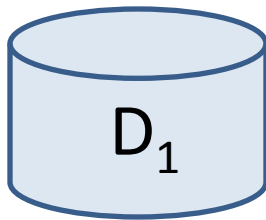


We can achieve what we want with a total of 9 disks. Can we do better?

---

# Another Example (III)

$$D = (D_1, D_2, D_3)$$

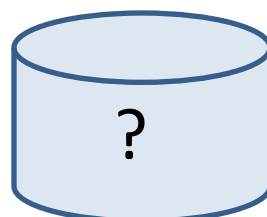
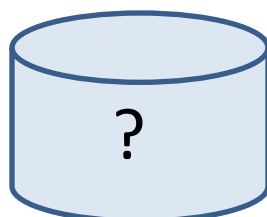
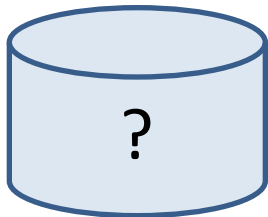


$$D_1 + D_2$$

$$D_1 + D_3$$

$$D_2 + D_3$$

$$D_1 + D_2 + D_3$$



To tolerate two failures, we need each  $D_i$  to be present at least 3 times.

---

# Summary

- Modulo  $n$
- Binary arithmetic
  - Binary vectors
  - Counting in binary
- Applications of binary arithmetic
  - storage

