Professor Hamlin
Day 7

Agenda
1) Review
2) Computer representation of sets
3) Set and Logic Algebra
4) Digital circuits

## Review

Sets: unordered collection of unique elements

Set builder notation, venn diagram

Set ops $A^c = \bar{A}$, $A \cup B$, $A \cap B$, $A - B$, $A \Delta B$

$\in$ , $\subseteq$ , $\subset$

cardinality, powersets $\mathbb{P}$

Exercise:

1) $A = \{\{1,2\}, 3, 4\}$ $B = \{1, 2, 3\}$

a) what is $A \cup B$? $1, 2 \neq \{1,2\}$

$\{\{1,2\}, 3, 4, 1, 2\}$

b) what is $|A \cap B|$? $\{3\}$

Bonus c) $\{1,2\} \in A - B$? $\{\{1,2\}, 4\}$

$\neq (2 \in A - B) \wedge (1 \in A - B)$

True

Bonus d) $\{1,2\} \subseteq A \Delta B$? $\to \{\{1,2\}, 1, 2, 4\}$ True

# Representing sets on Computers

Remember how we figured out power sets?

$A = \{a, b\}$

$\mathcal{P}(A) = \{\{a\}, \{b\}, \{a, b\}$ $\{b, a\}$? $\emptyset$?

| a | b | $\mathcal{P}(\{a,b\})$ |
|---|---|---|
| F | F | $\emptyset$ |
| F | T | $\{b\}$ |
| T | F | $\{a\}$ |
| T | T | $\{a, b\}$ |

Computers like things that are binary and fit in a set amount of bits
How can we do this for sets?

| 1<br>a | 2<br>b | $\mathcal{P}(A)$ | |
|---|---|---|---|
| F → 0 | F → 0 | $\emptyset$ | → 0 0 |
| F → 0 | T → 1 | $\{b\}$ | → 0 1 |
| T → 1 | F → 0 | $\{a\}$ | → 1 0 |
| T → 1 | T → 1 | $\{a, b\}$ | → 1 1 |


step 1 → assign index position to all elements in universe - bit string as long as universe

step 2 → if item 1 in set → index i set to 1
not in set → index i set to 0

e.g. $U = \{\text{green}, \text{black}, \text{red}, \text{white}, \text{blue}\}$

1 2 3 4 5

$A = \{\text{black}, \text{red}, \text{blue}\}$

0 1 1 0 1

1 0 0 1 1 $\Rightarrow \{\text{green}, \text{white}, \text{blue}\}$

## Connecting set operators to logic op

$U = \{\text{green}, \text{black}, \text{red}, \text{white}, \text{blue}\}$

1 2 3 4 5

$A = \{\text{black}, \text{red}, \text{blue}\}$

0 1 1 0 1

$B = \{\text{green}, \text{black}, \text{blue}\}$

1 1 0 0 1

$A \cup B = \{\text{green}, \text{black}, \text{red}, \text{blue}\}$

1 1 1 0 1

$$\begin{array}{r} \boxed{0}\,1101 \\ \text{or}\ \boxed{1}\,1001 \\ \hline \boxed{1}\,1101 \end{array}$$

| Sets | Logic | |
|---|---|---|
| $\bar{A} = A^c = \{\text{green}, \text{white}\}$ | $a = 01101$<br>$\neg a = 10010$ | each bit negated |
| $A \cup B$ | $a = 01101$<br>$b = 11001$<br>$a \vee b = 11101$ | apply or to each bit |

| | |
|---|---|
| $A \cap B = \{\text{black}, \text{blue}\}$ | $a = 01101$<br>$b = 11001$<br>$a \wedge b = 01001$<br>apply <u>and</u> to each bit |
| $A \triangle B = \{\text{green}, \text{red}\}$ | $a = 01101$<br>$b = 11001$<br>$a \text{ xor } b = 10100$<br>apply <u>xor</u> to each bit |

## Connecting set/logic operators

| Set | Logic |
|---|---|
| $^c$ (complement) | $\neg$ (negation) |
| $\cap$ (intersection) | $\wedge$ (and) |
| $\cup$ (union) | $\vee$ (or) |
| $\triangle$ (sym. diff.) | XOR |

# Set Algebra (Logic Algebra)

algebra ....

$$x(x+10)-2x+15$$

$$x^2+10x-2x+15$$

$$x^2+8x+15$$

manipulating expressions to simplify them.
Can do same for Logic / Set expressions!
We have a few rules to help us.

**Algebra**

$$(9+1)+5 = 9+(1+5)$$

$$(x+y)+z = x+(y+z)$$

**Logic**

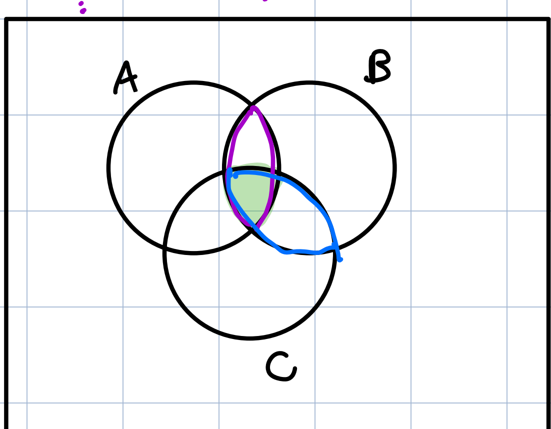
| $p$ | $q$ | $r$ | $(p \vee q) \vee r$ | $p \vee (q \vee r)$ |
|---|---|---|---|---|
| F | F | F | F | F |
| F | F | T | T | T |
| F | T | F | T | T |
| F | T | T | T | T |
| ⋮ |  | ⋮ | ⋮ | ⋮ |

**Associative Laws**

$(P \vee Q) \vee R = P \vee (Q \vee R)$

$(P \wedge Q) \wedge R = P \wedge (Q \wedge R)$

**Sets**




$$(A \cup B) \cup C = A \cup (B \cup C)$$

$$\underbrace{(A \cap B)} \cap C = A \cap \underbrace{(B \cap C)}$$

# Other Rules

**Double Negation**

$\neg\neg P = P$

$$(A^C)^C = A$$

$T \to F \to T$

**Distributive Laws**

$P \wedge (Q \vee R) = (P \wedge Q) \vee (P \wedge R)$

$P \vee (Q \wedge R) = (P \vee Q) \wedge (P \vee R)$

$$A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$$
$$A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$$

Idea: $x * (y + z) = x * y + x * z$

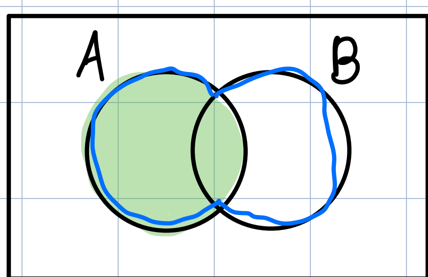
**Absorption Laws**

$P \wedge (P \vee Q) = P$

$P \vee (P \wedge Q) = P$

$$A \cap (A \cup B) = A$$
$$A \cup (A \cap B) = A$$

Idea:



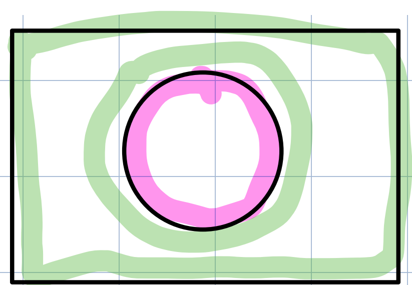

**Complement Laws**

$P \vee \neg P = T$

$P \wedge \neg P = F$

$$A \cup A^C = U$$
$$A \cap A^C = \emptyset$$

Idea:



**Idempotent Laws** (why so complicated?)

P ∨ P = P

P ∧ P = P

$A \cup A = A$

$A \cap A = A$

**Identity**

False ∨ P = P

True ∧ P = P

idea:

| P | P∨F | P∧T | P∧F | P∨T |
|---|---|---|---|---|
| F | F | F | F | T |
| T | T | T | F | T |

$\emptyset = \{\}$

$\emptyset \cup A = A$

$U \cap A = A$

universe

**Domination:**

True ∨ P = True

False ∧ P = False

$U \cup A = U$

$\emptyset \cap A = \emptyset$

**DeMorgan's Laws**

¬(P ∨ Q) = ¬ P ∧ ¬ Q

¬(P ∧ Q) = ¬ P ∨ ¬ Q

$(A \cup B)^C = A^C \cap B^C$

$(A \cap B)^C = A^C \cup B^C$

Recall from Day 4

Exercise Build Truth tables for

1) $\neg(A \vee B)$ 2) $\neg A \wedge \neg B$

| A | B | A∨B | ¬(A∨B) |
|---|---|---|---|
| F | F | F | T |
| F | T | T | F |
| T | F | T | F |
| T | T | T | F |

| A | B | ¬A | ¬B | ¬A∧¬B |
|---|---|---|---|---|
| F | F | T | T | T |
| F | T | T | F | F |
| T | F | F | T | F |
| T | T | F | F | F |

two statements are logically equivalent

# Simplifying Boolean or set expressions

Ex) $(X \cup Y) \cap (X \cup \bar{Y})$

(distributive law)

$X \cup (Y \cap \bar{Y})$

(complement law)

$X \cup \emptyset$

(identity law)

$\boxed{X}$

Ex) $\neg(\neg A \vee B) \wedge \neg B$

$(\neg\neg A \wedge \neg B) \wedge \neg B$ demorgan's

$(A \wedge \neg B) \wedge \neg B$ double negation

$A \wedge (\neg B \wedge \neg B)$ associative

$P \wedge P$ Idempotent

$A \wedge \neg B$

Practice 1) $(A \cup B) \cap \bar{A}$

$(\bar{A} \cap A) \cup (B \cap \bar{A})$ dist.

$\emptyset \cup (B \cap \bar{A})$ comp.

$B \cap \bar{A}$ identity

2) $(\neg x \wedge x) \vee (y \wedge \neg\neg x)$

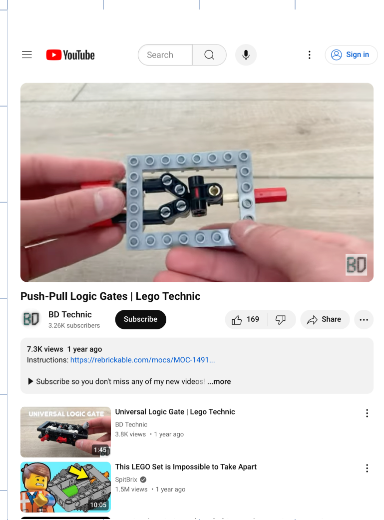
$(\neg x \wedge x) \vee (y \wedge x)$ double neg

$F \vee (y \wedge x)$ complement

$(y \wedge x)$ ident.

## Circuits on Computers

https://youtu.be/RA2po1xk_0A?t=5



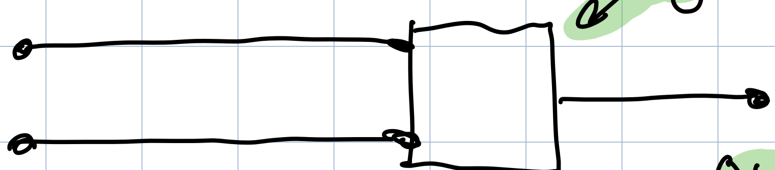

wire

electricity = 1

no electricty = 0

Can combine wires! This forms a.

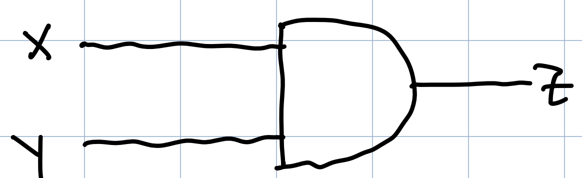
gate



Inputs

Output

Each of our boolean operators has a corresponding gate



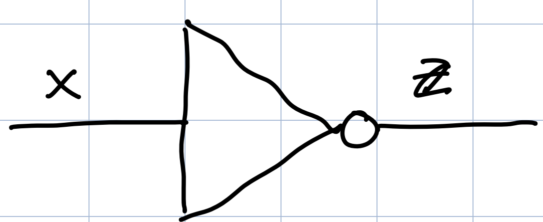

AND

| X | Y | Z |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

Note when talking about circuits we use 0/1 instead of T/F

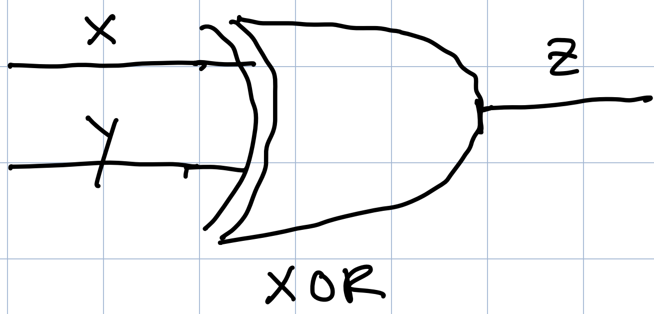

OR

| X | Y | Z |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |



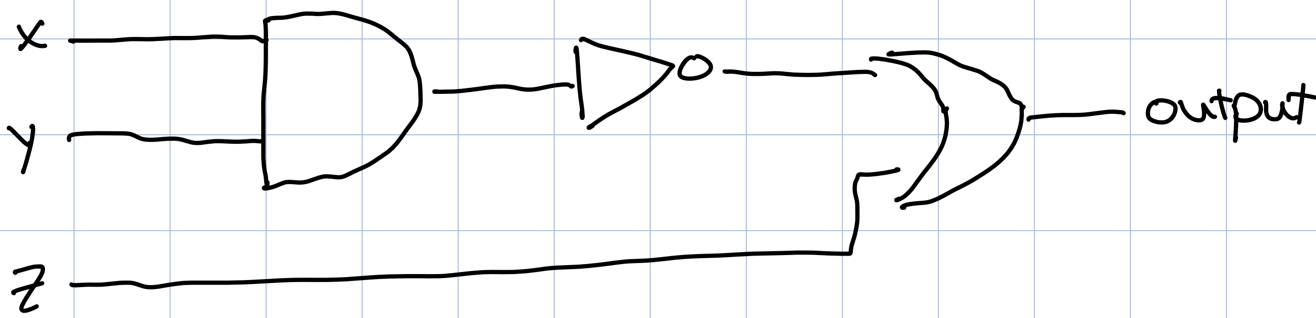

NOT

| X | Z |
|---|---|
| 0 | 1 |
| 1 | 0 |




| x | y | z |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

Circuits are when we connect gates....

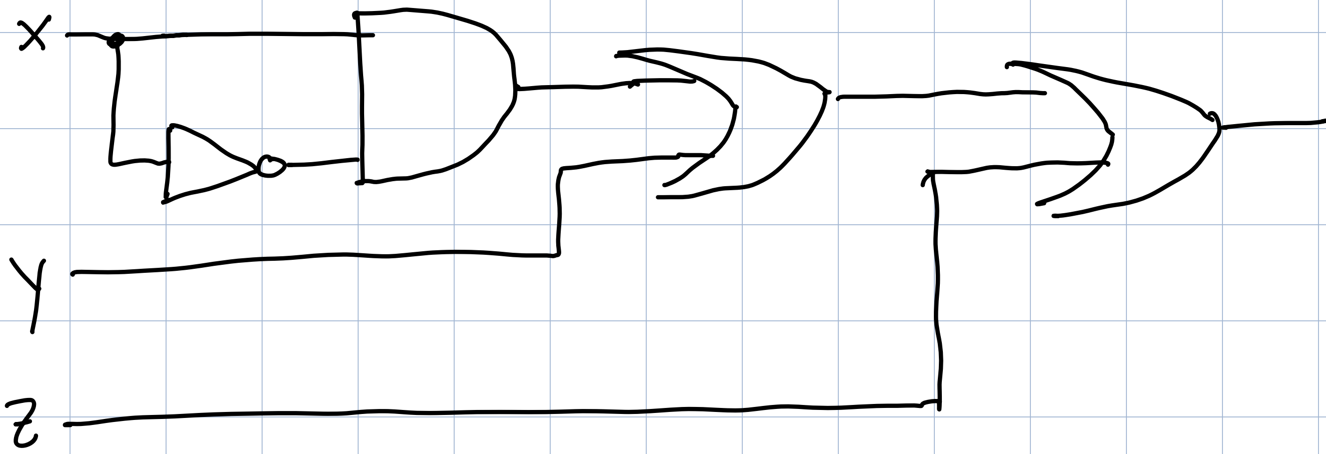



The logic expression above is...

$\neg(x \wedge y) \vee z$

Hint: work left to right when applying operations and remember your ()

## Exercise:




1) express using logic

$(x \wedge \neg x) \vee y \vee z$

2) simplify above expression using logic rules

$(x \wedge \neg x) \vee y \vee z$

$F \vee y \vee z$ complement

$y \vee z$ identity

3) Draw simplified expression