9/29/14

Exam 1 Total = 30

Average = 26.7

Quartiles = 25, 28, 29

St. Deviation = 3.6

Exam 2 will be Wed Oct 29

We never had a follow-up discussion for HW 2, so let's do that now.

We defined abstract functions $f: X \to Y$ but how does this agree with your previous experience?

Example: If X and Y are sets of numbers then we can define functions using "algebraic formulas".

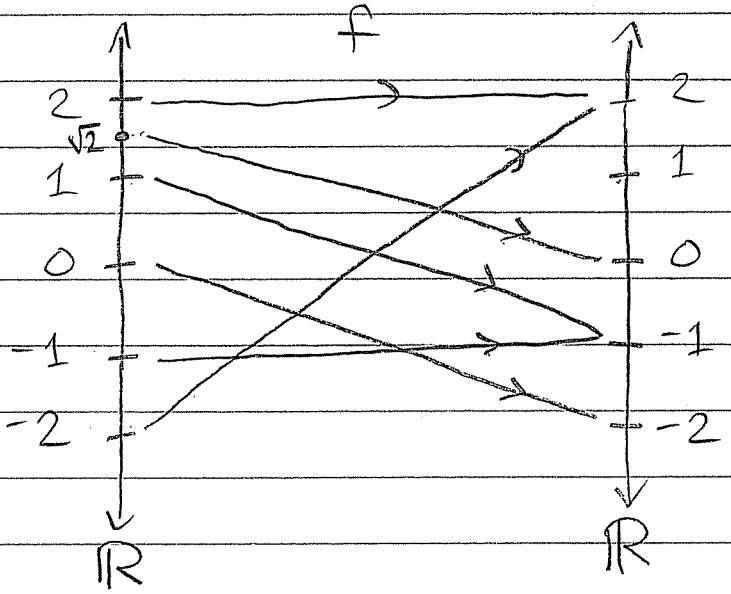
Recall the set of real numbers:

$\mathbb{R}$ = The set of numbers that have decimal expansions.

[Remark: It's difficult to define $\mathbb{R}$ formally, so we won't. See math 433 or 533 for this.]

The formula $f(x) = x^2 - 2$ defines a function $f: \mathbb{R} \to \mathbb{R}$.

How might we draw this function?



injective?
surjective?

There are too many arrows! I can't draw them all. Maybe there is a better way to represent this function?

We need another concept.

Definition: An ordered pair is a "set" with two elements in which order does matter. We will write it like this

$$(x, y)$$

x: 1st element y: 2nd element

[Remark: If you really want to express (x,y) using set language you can say something like

$$(x,y) := \{x, \{y\}\}.$$

But we won't bother with this extreme level of formality.]

Then given two sets A and B we define their Cartesian product

$$A \times B := \{(x,y) : x \in A \text{ and } y \in B\}.$$

Example: Let $A = \{a, b\}$ and $B = \{p, q, r\}$. Then the Cartesian product is

$$A \times B = \{(a,p), (a,q), (a,r), (b,p), (b,q), (b,r)\}$$

See the rectangle?

	p	q	r
a	(a,p)	(a,q)	(a,r)
b	(b,p)	(b,q)	(b,r)

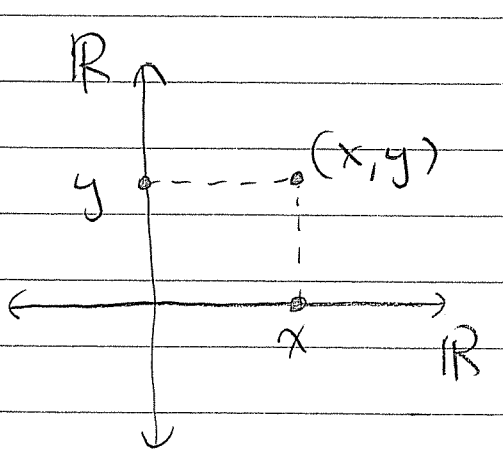
If A and B are finite sets then we have

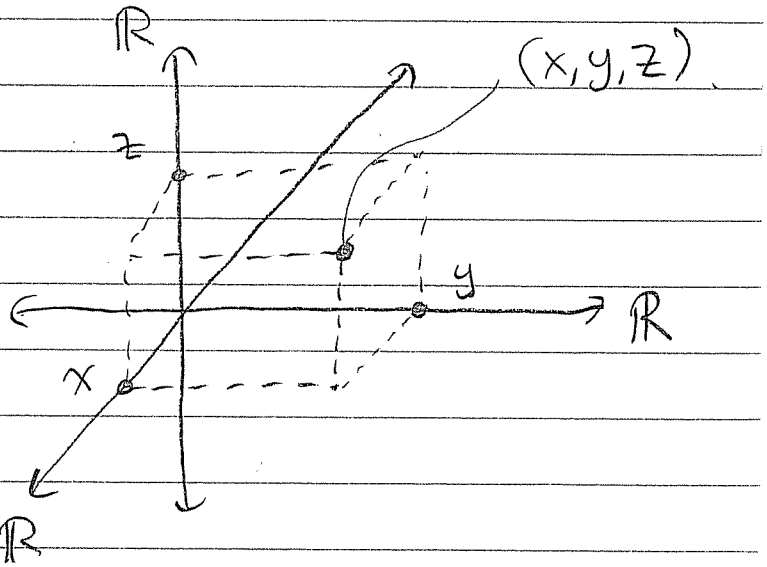
$$\#(A \times B) = \#A \times \#B .$$

(This explains the notation.)

Q: Why is this called the "Cartesian" product?

A: In 1637 René Descartes had the revolutionary idea that the sets $\mathbb{R}^2 := \mathbb{R} \times \mathbb{R}$ and $\mathbb{R}^3 := \mathbb{R}^2 \times \mathbb{R} = \mathbb{R} \times \mathbb{R} \times \mathbb{R}$ could be used to represent 2D and 3D space.





[Wait! Did we define "ordered triples"?
You can just say

$$(x,y,z) := ((x,y),z) \text{ or } (x,(y,z)).$$

It doesn't matter which. We say that the Cartesian product is "associative":

$$(A \times B) \times C \overset{\text{not really the same, but there is a natural bijection between them.}}{=} A \times (B \times C)$$

We'll just call this set $A \times B \times C$.]

Now we can define the "graph" of a function $f: A \to B$. To each arrow $x \longrightarrow f(x)$ we associate the ordered pair $(x, f(x)) \in A \times B$.

Definition: Let $f: A \to B$ be a function. Its graph is the following subset of $A \times B$

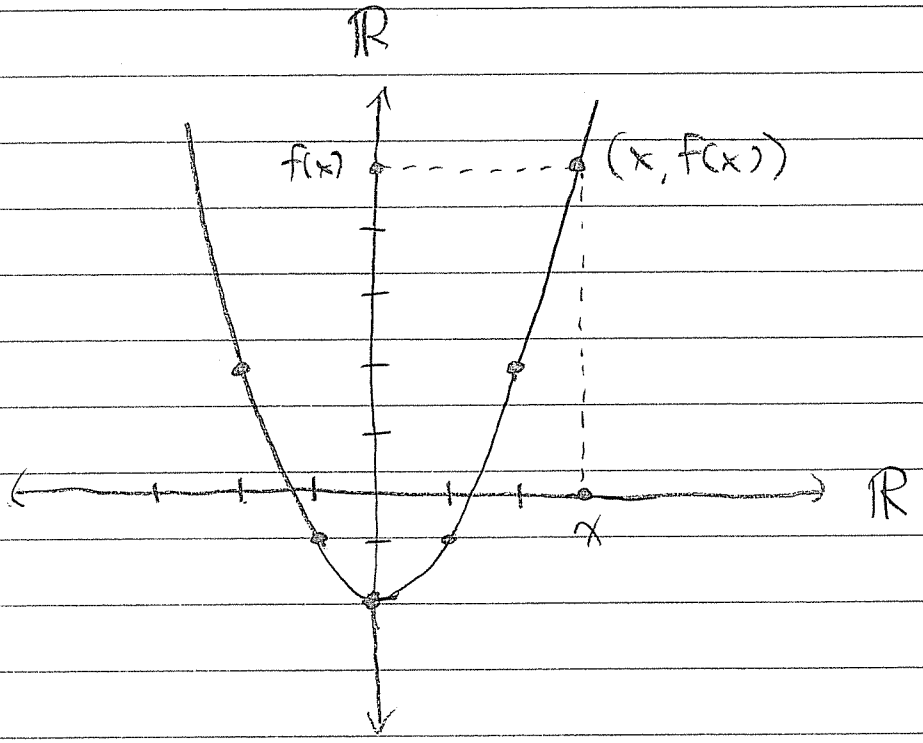
$$\{(x, f(x)) : x \in A\} \subseteq A \times B.$$

If the set $A \times B$ can be visualized, then this allows us to visualize functions $f: A \to B$ as certain subsets of $A \times B$.

Example: Recall the function $f: \mathbb{R} \to \mathbb{R}$ defined by $f(x) = x^2 - 2$. Its graph is the set

$$\{(x, x^2-2) : x \in \mathbb{R}\} \subseteq \mathbb{R} \times \mathbb{R} = \mathbb{R}^2,$$

which we can visualize as a "curve" in the cartesian plane $\mathbb{R}^2$.



This also allows us to rephrase the defining properties of a function.

Consider any subset $X \subseteq \mathbb{R}^2$. We say that X is a function if it satisfies two properties

(F1) Every vertical line intersects X at most once

(F2) Every vertical line intersects X at least once.

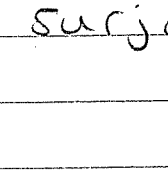
Additionally, we say X is injective if

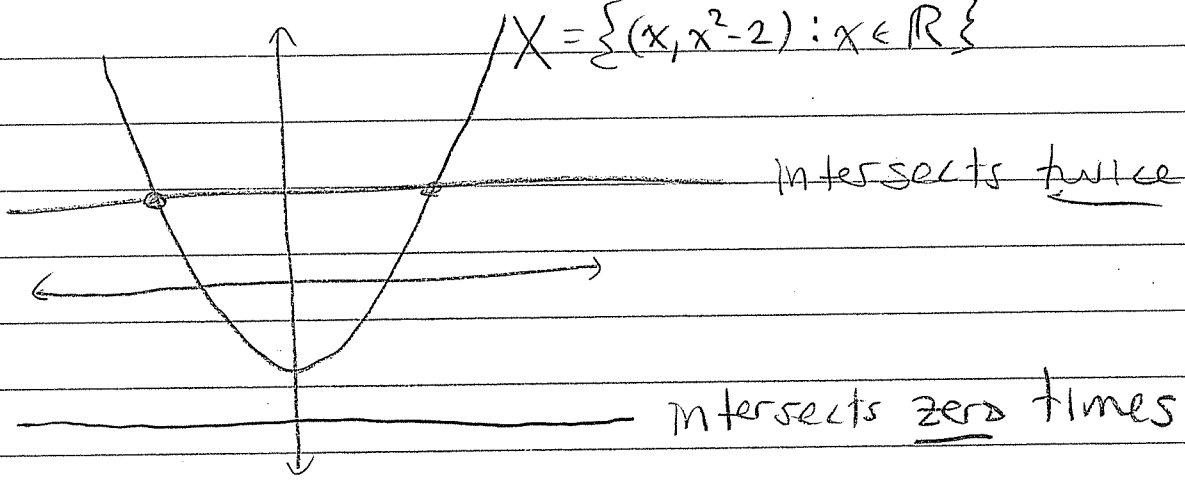
(F3) Every horizontal line intersects X at most once.

We say X is surjective if

(F4) Every horizontal line intersects X at least once.

Example: $f(x) = x^2 - 2$ is NOT injective and NOT surjective.





We say X is bijective / invertible if it intersects each horizontal line exactly once, in which case the inverse function is obtained by reflecting across the line $y = x$.

Functions in Logic:

We also use the language of functions in the study of logic.

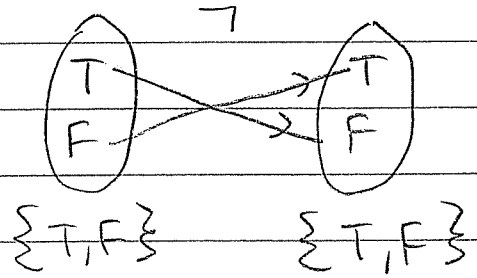
Definition: A Boolean function is any function of the form

$$f : \{T, F\}^n \to \{T, F\},$$

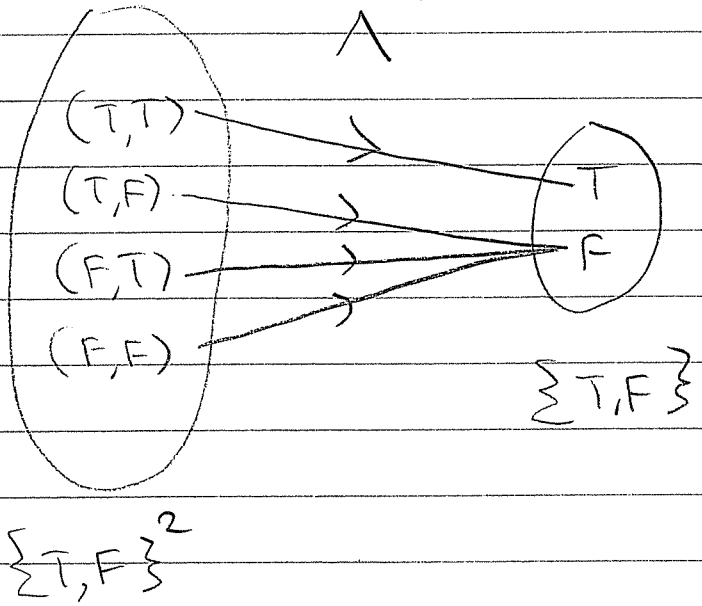
where $\{T,F\}^n$ is the set of "ordered n-tuples" of T's and F's.

We know three important examples:

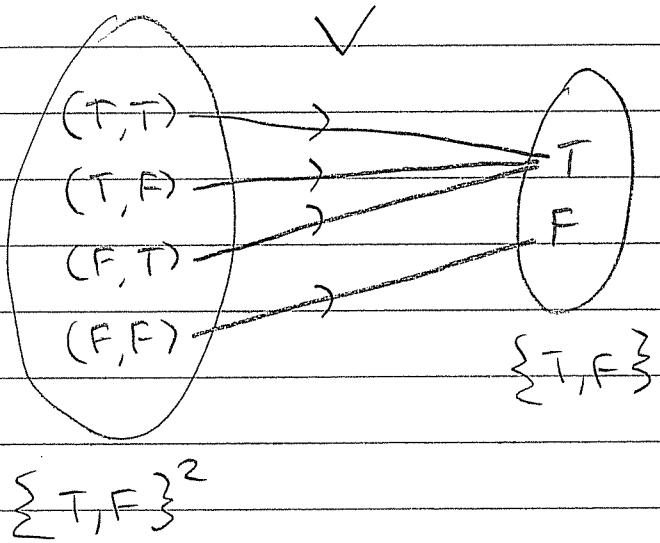
- $\neg : \{T,F\} \to \{T,F\}$



- $\wedge : \{T,F\}^2 \to \{T,F\}$



$V : \{T,F\}^2 \to \{T,F\}$



What do the graphs of these functions look like?

The graph of $\wedge : \{T,F\}^2 \to \{T,F\}$ is the set

$$\{((P,Q), P \wedge Q) : (P,Q) \in \{T,F\}^2\}$$

$$= \{((T,T),T), ((T,F),F), ((F,T),F), ((F,F),F)\},$$

which is a subset of

$$\{T,F\}^2 \times \{T,F\} \quad (= \{T,F\}^3)$$

we might as well just write it like this

Since the graph is just a finite set we will prefer to write it as a table:

P	Q	$P \wedge Q$
T	T	T
T	F	F
F	T	F
F	F	F

A truth table is just the graph of a Boolean function

Thinking Problem:

How many different Boolean functions are there in n variables?

$$f : \{T, F\}^n \longrightarrow \{T, F\}$$

10/1/14

HW 3 due next Wed Oct 8
Exam 2 Wed Oct 29

Last time we talked about functions; in particular, Boolean functions

$$\varphi : \{T,F\}^n \longrightarrow \{T,F\}$$

Recall that

$$\{T,F\}^n = \underbrace{\{T,F\} \times \{T,F\} \times \cdots \times \{T,F\}}_{(n \text{ times})}$$

= The set of ordered n-tuples of T's and F's.

Q: How many Boolean functions are there in n variables?

A: We know that the total # of functions

$$\varphi : \{T,F\}^n \longrightarrow \{T,F\}$$

equals $\#\{T,F\}^{\#(\{T,F\}^n)}$

↓

We also know that

$$\#(\{T,F\}^n) = (\#\{T,F\})^n$$

[Why? Because we have

$$\#(A \times B) = (\#A) \times (\#B)$$

for any finite sets A and B.]

Hence the total number of Boolean functions in n variables is

$$\#\{T,F\}^{(\#\{T,F\})^n} = 2^{2^n}.$$

That's a lot!

However, here is a remarkable fact:

Every Boolean function has an algebraic formula in terms of the three basis functions

$$\neg, \wedge, \vee.$$

To prove this fact let's go back to HW 2 Problem 4.

Q: Let U be a set with n elements. How many subsets does it have?

A: 2^n. Why?

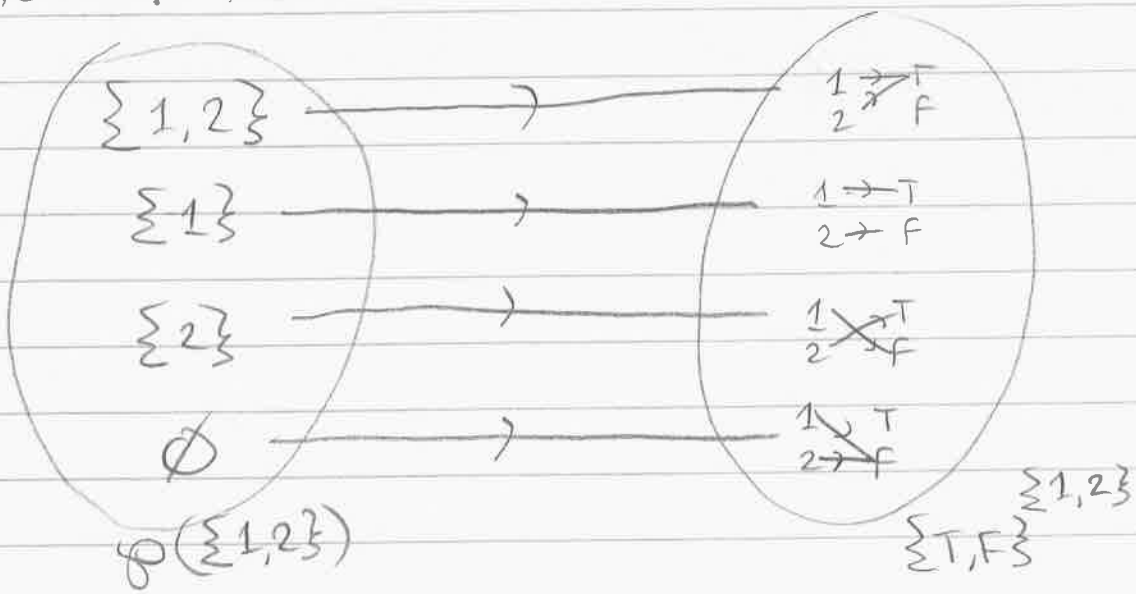
Let $\wp(U) :=$ The set of subsets of U.

Example: $\wp(\{1,2\}) = \{\{1,2\}, \{1\}, \{2\}, \emptyset\}$.

Let $\{T,F\}^U :=$ The set of functions $U \to \{T,F\}$.

Example: $\{T,F\}^{\{1,2\}} = \left\{ \begin{array}{l} 1 \to T \\ 2 \to F \end{array}, \begin{array}{l} 1 \to T \\ 2 \to F \end{array}, \begin{array}{l} 1 \to T \\ 2 \to F \end{array}, \begin{array}{l} 1 \to T \\ 2 \to F \end{array} \right\}$

Notice that there is a natural bijection between these sets



How can we describe this in general?

To each subset $A \subseteq U$ we associate a function $P_A : U \to \{T, F\}$ defined by

$$P_A(x) := \begin{cases} T & \text{if } x \in A \\ F & \text{if } x \notin A \end{cases}$$

We could even call this function

$$P_A(x) = \text{"}x \in A\text{"}.$$

And to each function $P : U \to \{T, F\}$ we associate a subset $U_P \subseteq U$ defined by

$$U_P := \{ x \in U : P(x) = T \}.$$

[These things are already familiar to you. We're just being a little abstract right now.]

The rule $A \to P_A$ defines a function $\wp(U) \to \{T, F\}^U$ and the rule $P \to U_P$ defines a function $\{T, F\}^U \to \wp(U)$.

These two functions are inverses of each other, so we obtain a bijection

$$\wp(U) \longleftrightarrow \{T,F\}^U$$

Who cares? Since $\exists$ a bijection, HW 2.1c says these sets have the same size:

$$\#\wp(U) = \#\left(\{T,F\}^U\right)$$
$$= \#\{T,F\}^{\#U}$$
$$= 2^{\#U}$$

In particular,

$$\#\wp(\{1,2,\dots,n\}) = 2^n.$$

//

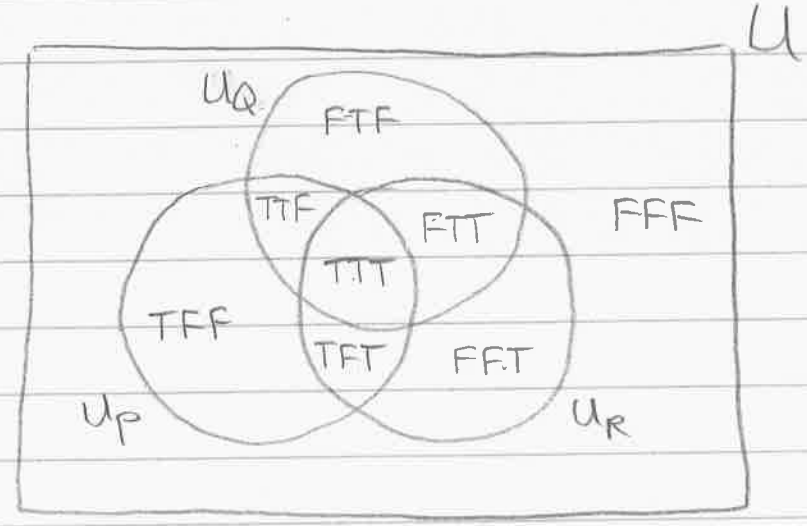
We can also use this correspondence to show that every Boolean function has a formula in terms of $\neg, \wedge, \vee$.

Example: Consider the following Boolean function $\varphi: \{T,F\}^3 \to \{T,F\}$ defined by

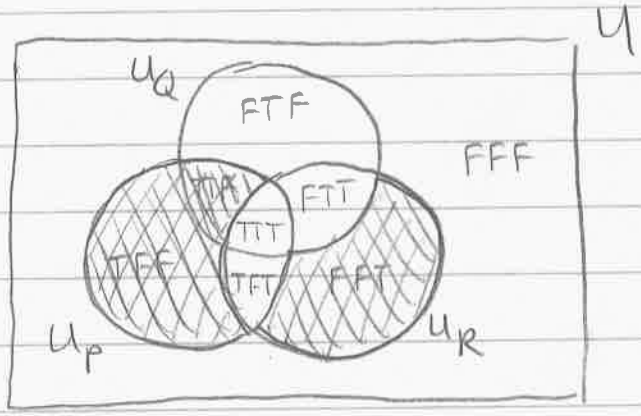
P	Q	R	$\varphi(P,Q,R)$
T	T	T	F
T	T	F	T
T	F	T	F
T	F	F	T
F	T	T	F
F	T	F	F
F	F	T	T
F	F	F	F.

Find a "formula" for $\varphi(P,Q,R)$.

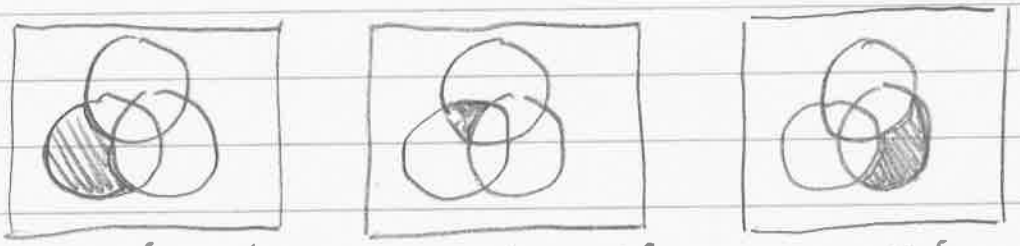
Solution: Let $U = \{T,F\}^3$. I can draw a nice picture of this set



The set $U_{\varphi(P,Q,R)}$ corresponds to the shaded region



Break the set into small pieces and put them back together



$U_P \cap U_Q^c \cap U_R^c$ $U_P \cap U_Q \cap U_R^c$ $U_P^c \cap U_Q^c \cap U_R$.

to get

$$U_{\varphi(P,Q,R)} = (U_P \cap U_Q^c \cap U_R^c) \cup (U_P \cap U_Q \cap U_R^c) \cup (U_P^c \cap U_Q^c \cap U_R)$$

Then apply the map $\varphi(U) \longrightarrow \{T,F\}^U$ to get back to Boolean functions :

$\varphi(P,Q,R) = (P \wedge Q \wedge \neg R) \vee (P \wedge \neg Q \wedge \neg R) \vee (\neg P \wedge \neg Q \wedge R)$

This expression is called the "disjunctive normal form" of φ.

It might not be the simplest formula but it has two advantages:

1. It is easy to compute.
2. We now have an algorithm to determine if two Boolean functions are equal: Put them both in disjunctive normal form and compare.

We have shown that every Boolean function has an "algebraic formula" in terms of $\neg, \wedge, \vee$.

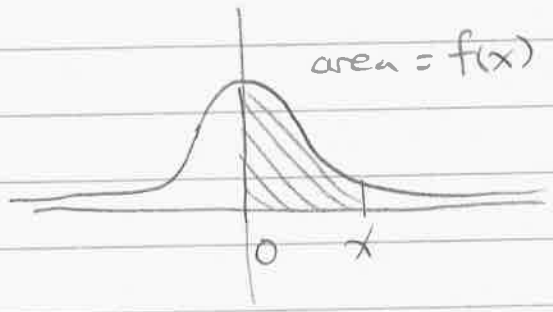
Contrast this with the case of real valued functions

$$f : \mathbb{R} \to \mathbb{R}.$$

The typical function $f: \mathbb{R} \to \mathbb{R}$ has no formula at all. Even some important functions have no algebraic formula

Consider the function

$$\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt$$



This function is vital in statistics, but it is a theorem that erf(x) cannot be expressed in terms of "elementary functions"

$$+, -, \times, \div, \exp, \log, \sin, \cos$$

So in this sense Boolean functions are much easier.

In fact, you will show on HW3 that every Boolean function can be expressed in terms of one single function called the

"Sheffer stroke"

For all $P, Q \in \{T, F\}$ we define

$$P \uparrow Q := \neg(P \wedge Q)$$

[We also call this the NAND function. Apparently, one type of flash memory (NAND flash memory) is based on the Sheffer stroke. There is also another kind of flash memory (NOR flash memory) based on "Peirce's arrow"

$$P \downarrow Q := \neg(P \vee Q) \quad .$$]

You will show that

$$\neg P = P \uparrow P$$

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

$$P \vee Q = (P \uparrow Q) \uparrow (P \uparrow Q) \quad .$$

Since every Boolean function can be expressed in terms of $\neg, \wedge, \vee$, now every Boolean function can be expressed in terms of $\uparrow$ alone.

Example: Express $\varphi(P,Q,R) = (P \wedge \neg Q) \vee R$ in terms of $\uparrow$ alone.

$$\varphi(P,Q,R) = (P \wedge (Q \uparrow Q)) \vee R$$

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

$$= \Big(\big((P \uparrow P) \uparrow ((Q \uparrow Q) \uparrow (Q \uparrow Q))\big) \uparrow R\Big)$$

$$\uparrow \Big(\big(\big((P \uparrow P) \uparrow ((Q \uparrow Q) \uparrow (Q \uparrow Q))\big) \uparrow R\big)\Big)$$

Nice, right?

Thinking Problem:

If you try to draw the Venn diagram for a Boolean function in 4 variables

$$\varphi : \{T,F\}^4 \to \{T,F\}$$

what happens? What is going on here?

Remark (culture):

IF we only allow ourselves to use the operations $\wedge$ and $\vee$, then NOT every Boolean function can be expressed.

The Boolean functions that CAN be so expressed are called monotone.

It is an open problem (called "Dedekind's problem", 1897) to find a formula for the number of monotone Boolean functions $\{T,F\}^n \to \{T,F\}$

10/6/14

HW 3 due this Wed.
Substitute Teacher next Mon Oct 13
and possibly on Mon Oct 20 also.
HW 4 due Mon Oct 27.
Exam 2 Wed Oct 29.

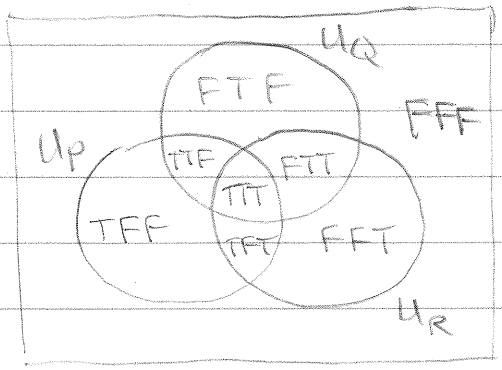
The next topic will be

The Binomial Theorem

but first I want to finish the story of Boolean Algebra.

We have seen that Venn diagrams and sets can be used to encode the same information as Truth tables and Boolean functions.

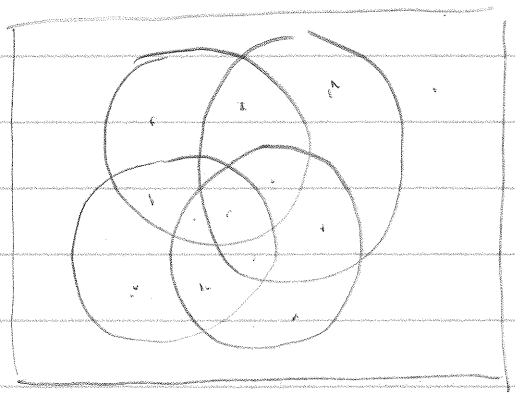
Example.



P	Q	R
T	T	T
T	T	F
T	F	T
T	F	F
F	T	T
F	T	F
F	F	T
F	F	F

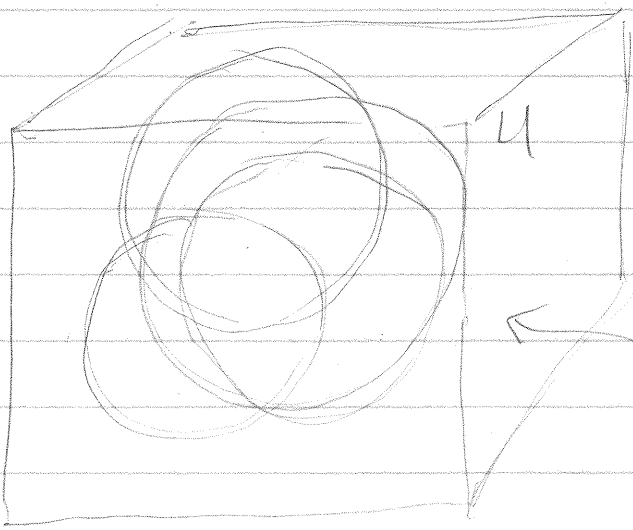
Thinking Problem:

What about a Boolean function in 4 variables, $\varphi: \{T,F\}^4 \to \{T,F\}$.
Can we encode it with a Venn diagram?



OOPS!
There are only 14 regions, but there should be 16 regions.

The correct Venn diagram should be 3-dimensional with 4 balls intersecting inside a box:

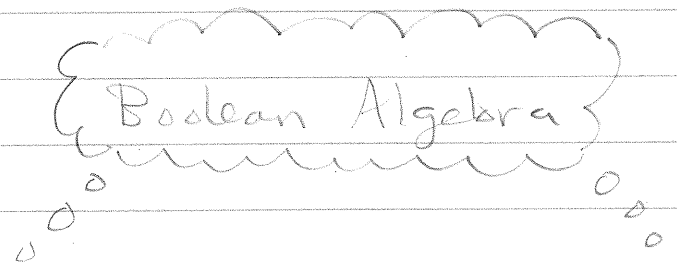


And this picture is not helpful to us humans.

Can you see the 16 regions?
Me neither.

Moral: Eventually we have to outgrow Venn diagrams

In the end we have to abandon specific interpretations and focus on the underlying structure.



This leads to a Definition:

An abstract Boolean Algebra is a structure $(B, \vee, \wedge, \neg, 0, 1)$ where

- B is a set.
- $\vee : B^2 \to B$ is a function ("join")
- $\wedge : B^2 \to B$ is a function ("meet")
- $\neg : B \to B$ is a function ("complement")
- $0, 1 \in B$ are special elements,

satisfying the following 5 rules/axioms:

(1) Associative Properties. $\forall a, b, c \in B$ we have

- $a \wedge (b \wedge c) = (a \wedge b) \wedge c$
- $a \vee (b \vee c) = (a \vee b) \vee c$

(2) Commutative Properties. $\forall a, b \in B$ we have

- $a \vee b = b \vee a$
- $a \wedge b = b \wedge a$

(3) Properties of 0 & 1. $\forall a \in B$ we have

- $a \vee 0 = a$
- $a \wedge 1 = a$

(4) Properties of Complement. $\forall a \in B$ we have

- $a \vee \neg a = 1$
- $a \wedge \neg a = 0$

(5) Distributive Properties. $\forall a, b, c \in B$ we have

- $a \vee (b \wedge c) = (a \vee b) \wedge (a \vee c)$
- $a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c)$

From these 5 axioms we can derive infinitely more true statements. Any true statement derived from the axioms is called a theorem. Note that we have a

☆ Duality Principle: Since the axioms are symmetric under simultaneously switching

$$\vee \leftrightarrow \wedge \quad \text{and} \quad 0 \leftrightarrow 1,$$

the theorems will come in "dual pairs". This will save us Lots of time.

[Remark: What motivates the definition of an abstract Boolean Algebra? By removing all human interpretation we make the language purely formal and suitable for computers. It also helps humans make fewer mistakes by converting arguments into the mechanical manipulation of symbols. (In my experience, students don't like arguments but they do like manipulation of symbols.)]

To illustrate how this works we will prove de Morgan's identities

- $\neg(a \vee b) = \neg a \wedge \neg b$
- $\neg(a \wedge b) = \neg a \vee \neg b$.

In a purely formal way.

(6) Theorem. $\forall\ a \in B$ we have

- $a \vee a = a$
- $a \wedge a = a$.

Proof: By Duality we only need to prove the first identity.

$$
\begin{aligned}
a &= a \vee 0 && (3)\\
&= a \vee (a \wedge \neg a) && (4)\\
&= (a \vee a) \wedge (a \vee \neg a) && (5)\\
&= (a \vee a) \wedge 1 && (4)\\
&= a \vee a && (3)
\end{aligned}
$$

□

Q.E.D.

///

etc.

(7) Theorem. We have

- $\neg 0 = 1$
- $\neg 1 = 0$

Proof: Again we just prove the first identity.

$$\begin{aligned} \neg 0 &= \neg 0 \vee 0 && (3) \\ &= 0 \vee \neg 0 && (2) \\ &= 1 && (4) \end{aligned}$$

//

(8) Theorem. $\forall a \in B$ we have

- $a \vee 1 = 1$
- $a \wedge 0 = 0$

Proof:

$$\begin{aligned} a \vee 1 &= a \vee (a \vee \neg a) && (4) \\ &= (a \vee a) \vee \neg a && (1) \\ &= a \vee \neg a && (6) \\ &= 1 && (4) \end{aligned}$$

//

[Wait! Are we allowed to use (6) in our proofs? It wasn't an axiom.
Yes. We are always allowed to use smaller numbers.]

(9) Absorption Properties. $\forall\, a, b \in B$ we have

- $a \wedge (a \vee b) = a$
- $a \vee (a \wedge b) = a$

Proof:

$$\begin{aligned} a \vee (a \wedge b) &= (a \wedge 1) \vee (a \wedge b) && (3) \\ &= a \wedge (1 \vee b) && (5) \\ &= a \wedge 1 && (2), (8) \\ &= a && (3) \end{aligned}$$

//

(10) Cancellation. Given $a, b, c \in B$. If $a \wedge c = b \wedge c$ and $a \vee c = b \vee c$, then it follows that $a = b$.

Proof: Assume that $a \wedge c = b \wedge c$ and $a \vee c = b \vee c$.

$a = a \vee (a \wedge c)$	(9)
$= a \vee (b \wedge c)$	hypothesis
$= (a \vee b) \wedge (a \vee c)$	(5)
$= (a \vee b) \wedge (b \vee c)$	hypothesis
$= b \vee (a \wedge c)$	(2), (5)
$= b \vee (b \wedge c)$	hypothesis
$= b$	(9)

//

(11) Uniqueness of Complements. Given $a, b \in B$. If $a \wedge b = 0$ and $a \vee b = 1$ then it follows that $b = \neg a$.

Proof: Assume $a \wedge b = 0$ and $a \vee b = 1$. Then

$a \wedge b = 0$	hypothesis
$= a \wedge \neg a$	(4)

and

$a \vee b = 1$	hypothesis
$= a \vee \neg a$	(4)

Then (10) implies $b = \neg a$. (How?) //

All that was preamble. Finally we can prove

(12) De Morgan's Identities. $\forall a,b,c \in B$,

- $\neg(a \vee b) = \neg a \wedge \neg b$
- $\neg(a \wedge b) = \neg a \vee \neg b$.

Proof: By (11) it is enough to show that $(a \vee b) \wedge (\neg a \wedge \neg b) = 0$ and $(a \vee b) \vee (\neg a \wedge \neg b) = 1$ to conclude $\neg a \wedge \neg b = \neg(a \vee b)$. Indeed,

$$
\begin{aligned}
&(a \vee b) \wedge (\neg a \wedge \neg b) \\
&\quad = [(\neg a \wedge \neg b) \wedge a] \vee [(\neg a \wedge \neg b) \wedge b] && (2), (5) \\
&\quad = [\neg b \wedge (a \wedge \neg a)] \vee [\neg a \wedge (b \wedge \neg b)] && (1), (2) \\
&\quad = [\neg b \wedge 0] \vee [\neg a \wedge 0] && (4) \\
&\quad = 0 \vee 0 && (5) \\
&\quad = 0 && (3) \text{ or } (6)
\end{aligned}
$$

and

$$
\begin{aligned}
&(a \vee b) \vee (\neg a \wedge \neg b) \\
&\quad = [(a \vee b) \vee \neg a] \wedge [(a \vee b) \vee \neg b] && (5) \\
&\quad = [b \vee (a \vee \neg a)] \wedge [a \vee (b \vee \neg b)] && (1), (2) \\
&\quad = [b \vee 1] \wedge [a \vee 1] && (4) \\
&\quad = 1 \wedge 1 && (8) \\
&\quad = 1 && (3) \text{ or } (6)
\end{aligned}
$$

//

DONE!

What was the point of that exercise?

We have shown that de Morgan's Laws are an inevitable consequence of axioms ① – ⑤. They do not depend on any kind of diagrams or human intuition.

Also, sometimes synthetic Boolean algebra is the most efficient solution to a problem.

Example: HW 3.4.

For all $a, b \in \mathbb{B}$ we define $a \uparrow b := \neg(a \wedge b)$. Show that

- $\neg a = a \uparrow a$
- $a \vee b = (a \uparrow a) \uparrow (b \uparrow b)$
- $a \wedge b = (a \uparrow b) \uparrow (a \uparrow b)$

Don't think! Just manipulate symbols.

10/8/14

No HW 4 yet.
Substitute teacher Mon Oct 13.
Class canceled Mon Oct 20.

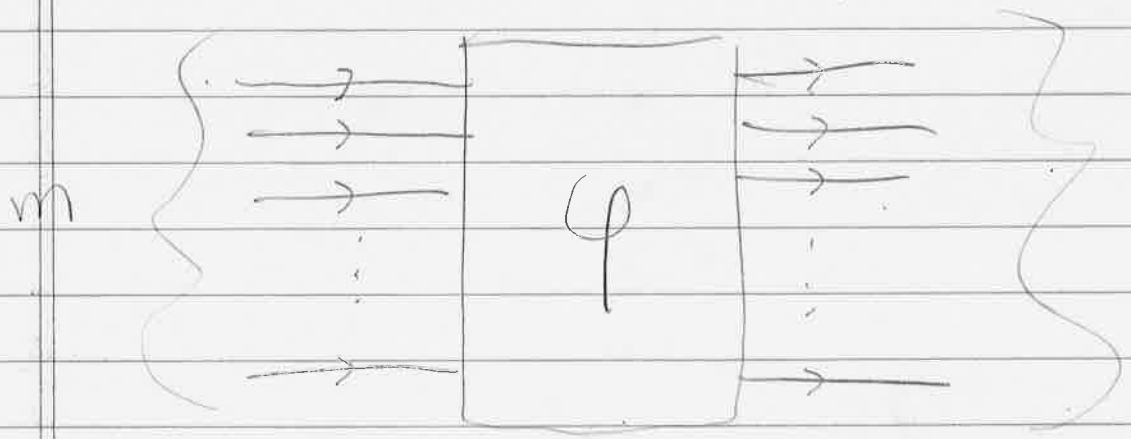
Today we will discuss HW3 and finish talking about Boolean algebra.

On Monday we will begin a new topic, the Binomial Theorem.

Why do you care about Boolean functions. In a nutshell, a Boolean function

$$\varphi : \{T, F\}^m \to \{T, F\}^n$$

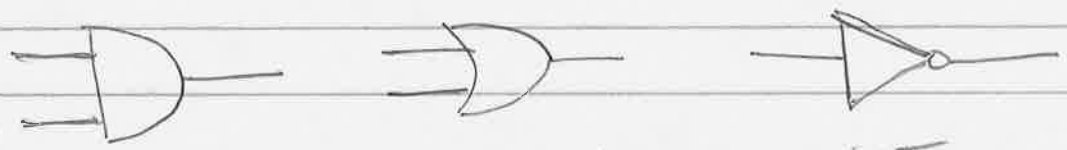
is just m wires going into a box and n wires coming out.



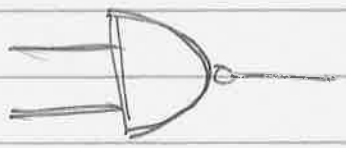
If we let T = 1 = high voltage
and F = 0 = low voltage
then φ just converts some currents in the input wires to currents in the output wires.

We would like to be able to build every possible function φ.

Because of the disjunctive normal form, it is always possible to build φ from the logic gates



If you can build these, you can build anything. On Problem 4 you proved that in fact you only need NAND gates



Let's do this together now.

First I need to prove one more theorem of Boolean Algebra.

(13) Theorem. Let B be a Boolean algebra. Then for all $a \in B$ we have

- $\neg(\neg a) = a$

Proof: To prove this we will use (11), the Uniqueness of Complements, to show that a equals the complement of $\neg a$. Indeed we have

$$\begin{aligned} \neg a \vee a &= a \vee \neg a && (2) \\ &= 1 && (4) \end{aligned}$$

and

$$\begin{aligned} \neg a \wedge a &= a \wedge \neg a && (2) \\ &= 0 && (4). \end{aligned}$$

Then (11) implies $a = \neg(\neg a)$. //

Now we prove

Problem 4: For all $a, b \in B$ we have

(a) $a \uparrow a = \neg a$

(b) $(a \uparrow a) \uparrow (b \uparrow b) = a \vee b$

(c) $(a \uparrow b) \uparrow (a \uparrow b) = a \wedge b$.

Proof: for (a) we have

$$\begin{aligned} a \uparrow a &= \neg(a \wedge a) && \text{definition} \\ &= \neg a. && (6) \end{aligned}$$

For (b) we have

$$\begin{aligned} (a \uparrow a) \uparrow (b \uparrow b) &= \neg a \uparrow \neg b && \text{part (a)} \\ &= \neg(\neg a \wedge \neg b) && \text{definition} \\ &= \neg\neg a \vee \neg\neg b && (12) \\ &= a \vee b. && (13) \end{aligned}$$

Finally, for (c) we have

$$\begin{aligned} (a \uparrow b) \uparrow (a \uparrow b) &= \neg((a \uparrow b) \wedge (a \uparrow b)) && \text{definition} \\ &= \neg(a \uparrow b) && (6) \\ &= \neg(\neg(a \wedge b)) && \text{definition} \\ &= a \wedge b. && (13) \end{aligned}$$

//

Now we can express any Boolean function in terms of the Sheffer stroke $\uparrow$ (we say it is a "universal" operation).

Example: Express $\varphi(a,b,c) = (a \wedge \neg b) \vee c$ in terms of $\uparrow$ alone.

$$(a \wedge \neg b) \vee c$$

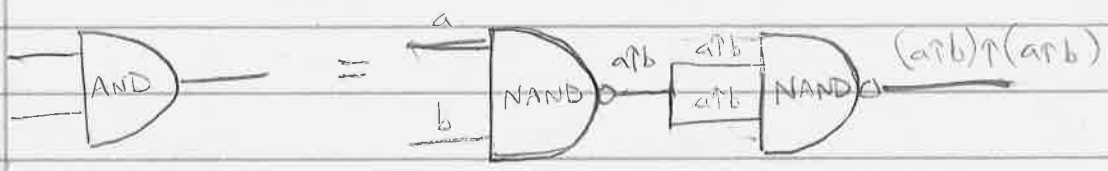
$$= (a \wedge (b \uparrow b)) \vee c$$

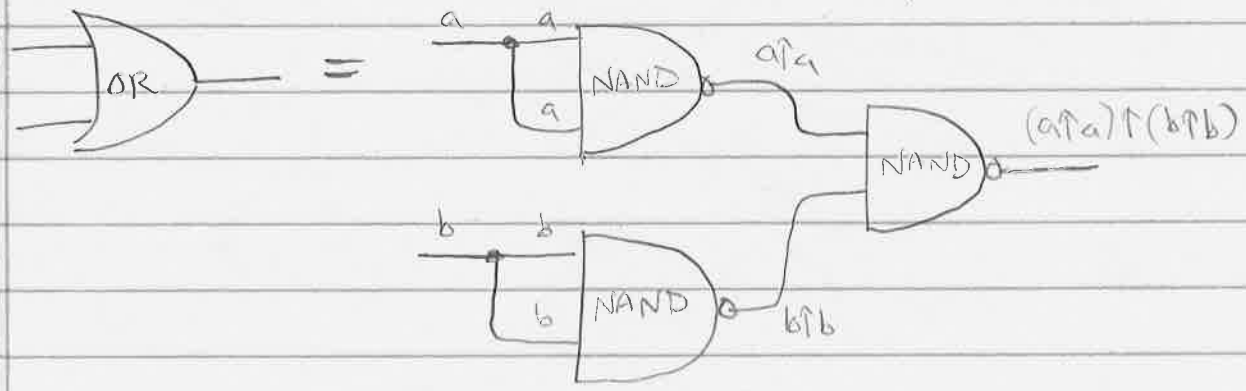
$$= ((a \uparrow (b \uparrow b)) \uparrow (a \uparrow (b \uparrow b))) \vee c$$

$$= [((a \uparrow (b \uparrow b)) \uparrow (a \uparrow (b \uparrow b))) \uparrow ((a \uparrow (b \uparrow b)) \uparrow (a \uparrow (b \uparrow b)))] \uparrow (c \uparrow c).$$

The formulas get long, as you see. This language is not good for humans but computers are happy with it. In fact, one kind of flash memory is built entirely out of NAND gates.

Heres how you can actually build AND, OR, NOT gates from NAND gates:





Here's a curious fact:

Our definition of Boolean Algebra used three operations $\neg, \wedge, \vee$
two special elements $0, 1$
and ten axioms.

Here is a surprising equivalent definition.

A Boolean algebra is a structure $(B, \uparrow)$ where B is a set and $\uparrow : B^2 \to B$ is a function satisfying just one axiom:

- $\forall\, a, b, c \in B$ we have

$$((a \uparrow b) \uparrow c) \uparrow (a \uparrow ((a \uparrow c) \uparrow a)) = c \,.$$

"Wolfram's axiom"

It would be a lot of work to show that these two definitions are equivalent (probably impossible to do by hand!).

It is also possible to define Boolean Algebra using just the two symbols

$$\wedge \text{ and } \oplus$$

because we can recover $\neg$ and $\vee$ from them.

Exercise: Show that $\forall a, b \in R$ we have

- $a \vee b = (a \oplus b) \oplus (a \wedge b)$
- $\neg a = 1 \oplus a$.

The most natural interpretation of the pair $\wedge, \oplus$ is via binary arithmetic.

If we define $1 = T$ and $0 = F$ then we have

$$a \wedge b = ab \mod 2$$
$$a \oplus b = a + b \mod 2.$$

Check:

a	b	$a \wedge b$	$a \oplus b$
1	1	$1 = 1 \cdot 1$	$0 = 1 + 1 \pmod 2$
1	0	$0 = 1 \cdot 0$	$1 = 1 + 0$
0	1	$0 = 0 \cdot 1$	$1 = 0 + 1$
0	0	$0 = 0 \cdot 0$	$0 = 0 + 0$.

From this it follows naturally that

$$a(b + c) = ab + ac$$
$$a \wedge (b \oplus c) = (a \wedge b) \oplus (a \wedge c).$$

Exercise: Is it true that

$$a \oplus (b \wedge c) = (a \oplus b) \wedge (a \oplus c) \quad ?$$

" $a + bc = (a+b)(a+b)$ "

$\Longrightarrow$

In summary, "Boolean Algebra" is a flexible abstract structure with several different concrete realizations.

① Set Theory.

Let U be a set and let

$B := \wp(U)$ = The set of all subsets of U.

Then B is a Boolean algebra with operations

$\vee$	$=$ $\cup$	union
$\wedge$	$=$ $\cap$	intersection
$\neg$	$=$ c	complement

and special elements

0	$=$ $\emptyset$	empty set
1	$=$ U	universal set.

//

(2) Logic

Let $\mathbb{B} = \{T, F\}$. Then $\mathbb{B}$ is a Boolean algebra with operations

$\vee$ = OR
$\wedge$ = AND
$\neg$ = NOT

and special elements $1 = T$, $0 = F$. //

(3) Binary Arithmetic.

Let $\mathbb{B} = \{0, 1\}$. Then $\mathbb{B}$ is a Boolean algebra with operations.

$\wedge = \cdot$ multiplication mod 2
$\oplus = +$ addition mod 2

and special elements $1 = 1$, $0 = 0$. //

We can choose the interpretation to suit our purpose.

To prepare you for the Binomial Theorem next week, let me mention one more realization of Boolean algebra.

Consider the set

$$U = \{1, 2, 3, \ldots, n\}$$

To each subset $A \subseteq U$ we associate a "binary string"

$$b_1 b_2 b_3 \cdots b_n$$

defined by $b_i = \begin{cases} 1 & \text{if } i \in A \\ 0 & \text{if } i \notin A. \end{cases}$

Example: The subset

$$\{2, 3, 5\} \subseteq \{1, 2, 3, 4, 5, 6\}$$

corresponds to the binary string

$$\begin{matrix} 0 & 1 & 1 & 0 & 1 & 0 \\ (1 & 2 & 3 & 4 & 5 & 6) \end{matrix}.$$

Example: The subsets of $\{1,2,3\}$ are

111

110 101 011

100 010 001

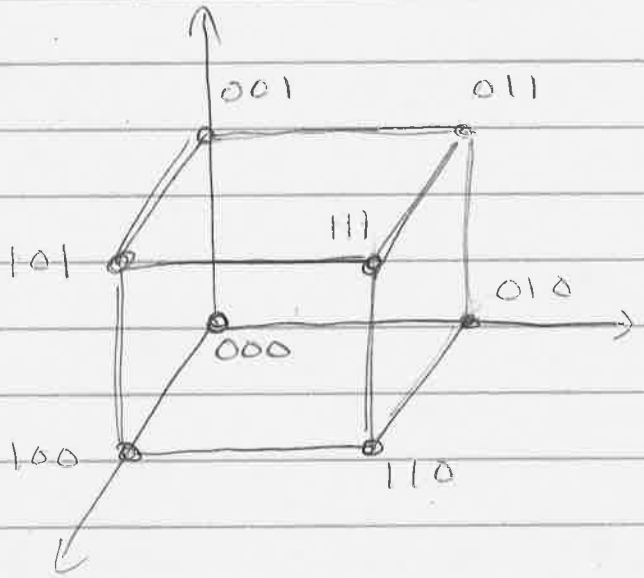
000 ///

Note that $\{1,2,3\}$ has 8 subsets because there are 8 binary strings of length 3:

$$\underbrace{2}_{1st} \times \underbrace{2}_{2nd} \times \underbrace{2}_{3rd} = 8$$

What if we think of binary strings as vectors and plot them in Cartesian space?

{



A Cube!

The subsets of $\{1,2,3\}$ are the same as the vertices of a cube.