Propositional Logic

CMSC 250

Reminders

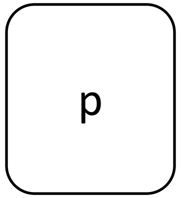
- Midterm in a week, 03 – 03
  - Announcement of 02-17 details schedule.
  - Material: everything from start through Thursday 2-25.
- Grade breakdown reflects weeks of coverage ☺
  - About 80% combinatorics, 20% logic

Propositional Logic

- The most elementary kind of logic in Computer Science
- Also known as Boolean Logic, by virtue of *George Boole* (1815 – 1864)

Propositional Symbols

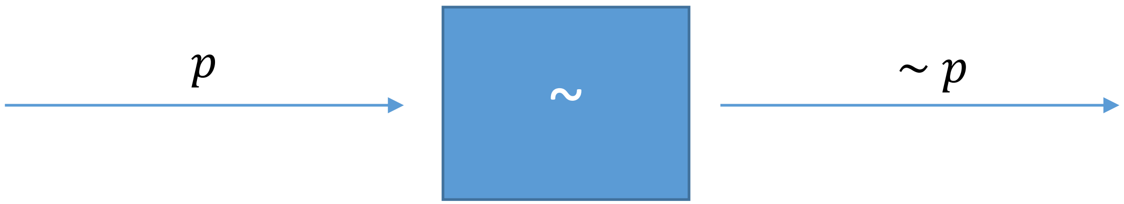
- The building blocks of propositional logic.
- Think of them as **bits** or **boxes** that hold a value of 1 (True) or 0 (False)
- Denoted using a lowercase english letter (p, q, … , a)



Operations in boolean logic

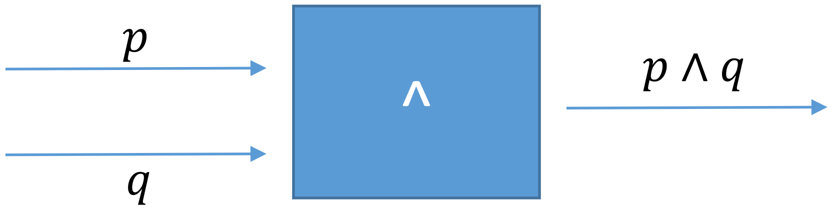
- There are three basic operations in boolean logic
  - Conjunction (AND)
  - Disjunction (OR)
  - Negation (NOT)
- Other operations can be defined *in terms of those three.*

Negation $(\mathrm{NOT}, \sim, \neg)$



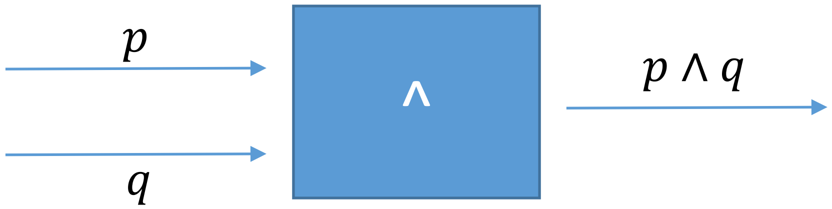
p	$\sim p$
F	***T***
T	***F***

Conjunction (^)



p	q	$p \wedge q$
F	***F***	***F***
F	***T***	***F***
T	***F***	***F***
T	***T***	***T***

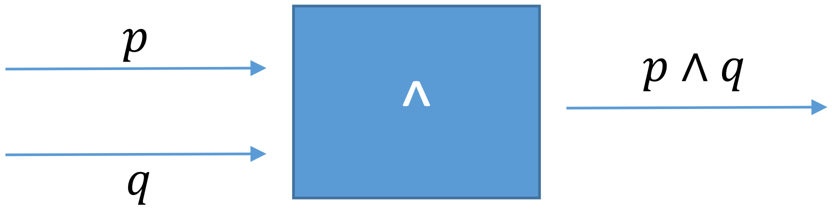
Conjunction (^)



p	q	$p \wedge q$
F	***F***	***F***
F	***T***	***F***
T	***F***	***F***
T	***T***	***T***

Rule of thumb: p **and** q must be 1

Conjunction (^)



p	q	$p \land q$	$q \land p$
F	***F***	***F***	***F***
F	***T***	***F***	***F***
T	***F***	***F***	***F***
T	***T***	***T***	***T***

Conjunction is commutative!

Fun exercise

- Fill-in the following truth table:

p	q	$p \wedge (\sim q)$
F	***F***	***?***
F	***T***	***?***
T	***F***	***?***
T	***T***	***?***

Fun exercise

- Fill-in the following truth table:

p	q	$p \wedge (\sim q)$
F	***F***	
F	***T***	
T	***F***	
T	***T***	

Fun exercise

- Fill-in the following truth table:

p	q	$p \wedge (\sim q)$
F	***F***	***F***
F	***T***	
T	***F***	
T	***T***	

Fun exercise

- Fill-in the following truth table:

p	q	$p \wedge (\sim q)$
F	*F*	*F*
F	*T*	*F*
T	*F*	
T	*T*	

Fun exercise

- Fill-in the following truth table:

p	q	$p \wedge (\sim q)$
F	***F***	***F***
F	***T***	***F***
T	***F***	***T***
T	***T***	

Fun exercise

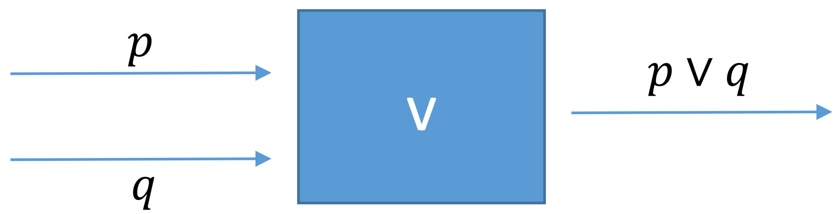
- Fill-in the following truth table:

p	q	$p \wedge (\sim q)$
F	*F*	*F*
F	*T*	*F*
T	*F*	*T*
T	*T*	*F*

Disjunction

p	q	$p \vee q$
F	***F***	***F***
F	***T***	***T***
T	***F***	***T***
T	***T***	***T***

Disjunction



p	q	$p \vee q$
F	***F***	***F***
F	***T***	***T***
T	***F***	***T***
T	***T***	***T***

Rule of thumb:
one of p **or** q
must be 1

Fun exercise

- Fill-in the following truth table:

p	q	$p \vee (p \wedge q)$
F	***F***	***?***
F	***T***	***?***
T	***F***	***?***
T	***T***	***?***

Fun exercise

- Fill-in the following truth table:

p	q	$p \vee (p \wedge q)$
F	***F***	
F	***T***	
T	***F***	
T	***T***	

Fun exercise

- Fill-in the following truth table:

p	q	$p \vee (p \wedge q)$
F	***F***	***F***
F	***T***	
T	***F***	
T	***T***	

Fun exercise

- Fill-in the following truth table:

p	q	$p \vee (p \wedge q)$
F	*F*	*F*
F	*T*	*F*
T	*F*	
T	*T*	

Fun exercise

- Fill-in the following truth table:

p	q	$p \vee (p \wedge q)$
F	***F***	***F***
F	***T***	***F***
T	***F***	***T***
T	***T***	

Fun exercise

- Fill-in the following truth table:

p	q	$p \vee (p \wedge q)$
F	*F*	*F*
F	*T*	*F*
T	*F*	*T*
T	*T*	*T*

Fun exercise

- Fill-in the following truth table:

p	q	$p \vee (p \wedge q)$
F	***F***	***F***
F	***T***	***F***
T	***F***	***T***
T	***T***	***T***

- Anything interesting here?

Fun exercise

- Fill-in the following truth table:

p	q	$p \vee (p \wedge q)$
F	*F*	*F*
F	*T*	*F*
T	*F*	*T*
T	*T*	*T*

- Anything interesting here?

Implication ($\Longrightarrow$)

- “If –then”

p	q	$p \Rightarrow q$
F	***F***	***T***
F	***T***	***T***
T	***F***	***F***
T	***T***	***T***

Implication ($\Longrightarrow$)

- "If –then"

p	q	$p \Rightarrow q$
F	*F*	*T*
F	*T*	*T*
T	*F*	*F*
T	*T*	*T*

Gorslax learns about birds

- Gorslax, an alien from the Andromeda Galaxy, visits planet Earth on a scientific expedition.
- Gorslax's planet has a very strong gravitational field which does not allow for the evolution of aviary life.
  - So he starts studying Earth's birds.

Gorslax learns about birds




Gorslax learns about birds

Hmmm… **From what I see until now**, it seems as if **all birds fly**

$bird$	$flies$	$bird \Rightarrow flies$
F	***F***	***T***
F	***T***	***T***
T	***F***	***F***
T	***T***	***T***

Gorslax learns about birds

Well this thing clearly doesn't fly, but it's also not a bird, so I don't care; I still believe that all birds fly!

$bird$	$flies$	$bird \Rightarrow flies$
F	*F*	*T*
F	*T*	*T*
T	*F*	*F*
T	*T*	*T*

Gorslax learns about birds

$bird$	$flies$	$bird \Rightarrow flies$
F	*F*	*T*
F	*T*	*T*
T	*F*	*F*
T	*T*	*T*

Gorslax learns about birds

Whoops! **Here's at least one bird that doesn't fly**! So my syllogism ***"if bird then flies"*** does not **universally** apply!

$bird$	$flies$	$bird \Rightarrow flies$
F	***F***	***T***
F	***T***	***T***
T	***F***	***F***
T	***T***	***T***

Bi-conditional ($\Leftrightarrow$)

- "If and only if"

p	q	$p \Leftrightarrow q$
F	*F*	*T*
F	*T*	*F*
T	*F*	*F*
T	*T*	*T*

Practice

- Fill in the following truth tables:

p	$(\sim p)$	$p \Rightarrow (\sim p)$
F	***T***	***?***
T	***F***	***?***

p	q	r	$p \wedge q$	$(p \wedge q) \Rightarrow r$	$(p \wedge q) \Leftrightarrow r$
F	***F***	***F***	***F***	***?***	***?***
F	***F***	***T***	***F***	***?***	***?***
F	***T***	***F***	***F***	***?***	***?***
F	***T***	***T***	***F***	***?***	***?***
T	***F***	***F***	***F***	***?***	***?***
T	***F***	***T***	***F***	***?***	***?***
T	***T***	***F***	***T***	***?***	***?***
T	***T***	***T***	***T***	***?***	***?***

Contradictions / Tautologies

- Examine the statements:
  - $p \wedge (\sim p)$
  - $p \vee (\sim p)$
- What can you say about those statements?

Another important equivalence

- Let's fill in the following truth table :

$\boldsymbol{a}$	$\boldsymbol{b}$	$\sim(\boldsymbol{a} \wedge \boldsymbol{b})$	$(\sim \boldsymbol{a}) \vee (\sim \boldsymbol{b})$
F	***F***	**?**	**?**
F	***T***	**?**	**?**
T	***F***	**?**	**?**
T	***T***	**?**	**?**

Another important equivalence

- Let's fill in the following truth table :

$\boldsymbol{a}$	$\boldsymbol{b}$	$\sim(\boldsymbol{a} \wedge \boldsymbol{b})$	$(\sim \boldsymbol{a}) \vee (\sim \boldsymbol{b})$
F	***F***	*T*	*T*
F	***T***	*T*	*T*
T	***F***	*T*	*T*
T	***T***	*F*	*F*

Another important equivalence

- Let’s fill in the following truth table :

a	b	$\sim (a \wedge b)$	$(\sim a) \vee (\sim b)$
F	*F*	*T*	*T*
F	*T*	*T*	*T*
T	*F*	*T*	*T*
T	*T*	*F*	*F*

- **These columns are the same!**
- **Conclusion:** $\sim (a \wedge b) \equiv (\sim a) \vee (\sim b)$

Another important equivalence

- Let's fill in the following truth table :

$\boldsymbol{a}$	$\boldsymbol{b}$	$\sim (\boldsymbol{a} \wedge \boldsymbol{b})$	$(\sim \boldsymbol{a}) \vee (\sim \boldsymbol{b})$
F	***F***	*T*	*T*
F	***T***	*T*	*T*
T	***F***	*T*	*T*
T	***T***	*F*	*F*

This result is known as
De Morgan's law

- **These columns are the same!**
- **Conclusion:** $\sim (a \wedge b) \equiv (\sim a) \vee (\sim b)$

Understanding De Morgan’s Law

- $\sim(\text{“}\mathit{Alice\ is\ Blonde}\text{”} \wedge \text{“}\mathit{Alice\ wears\ Green\ Dress}\text{”})$: **Clearly true**

Understanding De Morgan's Law

- $\sim(\text{"}Alice\ is\ Blonde\text{"} \wedge \text{"}Alice\ wears\ Green\ Dress\text{"})$: **Clearly true**

- $(\sim\text{"}Alice\ is\ Blonde\text{"}) \vee (\sim\text{"}Alice\ wears\ Green\ Dress\text{"})$:
  Also true!

De Morgan’s Laws (there’s two of them)

$$\sim (a \vee b) \equiv (\sim a) \wedge (\sim b)$$

$$\sim (a \wedge b) \equiv (\sim a) \vee (\sim b)$$

- Conjunctions flipped to disjunctions, and vice versa
- Negation operator ($\sim$) distributed across terms
- These laws give us our first pair of equivalent expressions!

Proving equivalences

- How do we prove an equivalence? (e.g $\sim(a \wedge b) \equiv (\sim a) \ \vee (\sim b)$)

Proving equivalences

- How do we prove an equivalence? (e.g $\sim(a \land b) \equiv (\sim a) \lor (\sim b)$)

1. Truth tables
   - One major problem: for n variables, 2^n rows (input combinations) to enumerate!

Proving equivalences

- How do we prove an equivalence? (e.g $\sim(a \wedge b) \equiv (\sim a) \vee (\sim b)$)

1. Truth tables
   - One major problem: for n variables, 2^n rows (input combinations) to enumerate!
   - Can we do better?

Proving equivalences

- How do we prove an equivalence? (e.g $\sim(a \wedge b) \equiv (\sim a) \vee (\sim b)$)

1. Truth tables
   - One major problem: for n variables, 2^n rows (input combinations) to enumerate!
   - Can we do better?
2. Laws of logical equivalence in a chain, one after the other!
   - We no longer have to compare 2^n input combinations to ensure that they all map to the same truth value (**T** or **F**). ☺
   - But somebody needs to code the system up!

Boolean Logic Cheat Sheet

Commutativity of binary operators	$p \wedge q \equiv q \wedge p$	$p \vee q \equiv q \vee p$
Associativity of binary operators	$(p \wedge q) \wedge r \equiv p \wedge (q \wedge r)$	$(p \vee q) \vee r \equiv p \vee (q \vee r)$
Distributivity of binary operators	$p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$	$p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$
Identity laws	$p \wedge T \equiv p$	$p \vee F \equiv p$
Negation laws	$p \vee ({\sim}p) \equiv T$	$p \wedge ({\sim}p) \equiv F$
Double negation	${\sim}({\sim}p) \equiv p$	
Idempotence	$p \wedge p \equiv p$	$p \vee p \equiv p$
De Morgan's axioms	${\sim}(p \wedge q) \equiv ({\sim}p) \vee ({\sim}q)$	${\sim}(p \vee q) \equiv ({\sim}p) \wedge ({\sim}q)$
Universal bound laws	$p \vee T \equiv T$	$p \wedge F \equiv F$
Absorption laws	$p \vee (p \wedge q) \equiv p$	$p \wedge (p \vee q) \equiv p$
Negations of contradictions / tautologies	${\sim}F \equiv T$	${\sim}T \equiv F$
Contrapositive	$(a \Rightarrow b) \equiv (({\sim}b) \Rightarrow ({\sim}a))$	
Equivalence between biconditional and implication	$a \Leftrightarrow b \equiv (a \Rightarrow b) \wedge (b \Rightarrow a)$	
Equivalence between implication and disjunction	$a \Rightarrow b \equiv {\sim}a \vee b$	

Proving equivalences using laws

- Suppose we want to investigate if

$$(((a \wedge b) \vee q) \wedge (b \wedge a)) \equiv (p \vee \sim p) \wedge ((a \wedge b) \vee ((\sim r) \wedge r))$$

- How many rows would the truth table have?

Proving equivalences using laws

- Suppose we want to investigate if

$$(((a \wedge b) \vee q) \wedge (b \wedge a)) \equiv (p \vee \sim p) \wedge ((a \wedge b) \vee ((\sim r) \wedge r))$$

- How many rows would the truth table have?
  - $2^5 = 32$ ☹ Too much time!

Proving equivalences using laws

- Suppose we want to investigate if

$$(((a \land b) \lor q) \land (b \land a)) \equiv (p \lor \sim p) \land ((a \land b) \lor ((\sim r) \land r))$$

- How many rows would the truth table have?
  - $2^5 = 32$ ☹ Too much time!

- Let's see how we could use the laws of logical equivalence to prove this equivalence. We will **document all laws except commutativity / associativity.**

More equivalences

- Please prove the following equivalences true or false!

$$a \Rightarrow b \equiv (\sim b) \Rightarrow (\sim a)$$

$$a \Rightarrow b \equiv (\sim a) \Rightarrow (\sim b)$$

$$a \Leftrightarrow b \equiv \big((\sim a) \vee b\big) \wedge ((\sim b) \vee a)$$

More equivalences

- Please prove the following equivalences true or false!

$$a \Rightarrow b \equiv (\sim b) \Rightarrow (\sim a) \quad \textit{(Contrapositive)}$$

$$a \Rightarrow b \equiv (\sim a) \Rightarrow (\sim b) \quad \textit{(Inverse Error)}$$

$$a \Leftrightarrow b \equiv \big((\sim a) \vee b\big) \wedge ((\sim b) \vee a)$$

Simplifying expressions

- Large expressions can often be **simplified** using the equivalences we discussed earlier.
- Example: Let's simplify $p \wedge (p \vee q) \wedge (p \wedge q)$

Simplifying expressions

- Large expressions can often be **simplified** using the equivalences we discussed earlier.
- Example: Let's simplify $p \land (p \lor q) \land (p \land q)$

Here's one way
$p \land (p \lor q) \land (p \land q)$ *(Original expression)* $\equiv p \land (p \land q)$ *(How?)* $\equiv (p \land p) \land q$ *(How?)* $\equiv p \land q$ *(How?)*

Your turn, class!

- Let's simplify the following three expressions.

$(a_1 \vee a_1) \wedge (a_2 \vee a_2) \wedge \cdots \wedge (a_{100} \vee a_{100}) \wedge (\sim a_1 \vee \sim a_1) \wedge (\sim a_2 \vee \sim a_2) \wedge \cdots \wedge (\sim a_{100} \vee \sim a_{100})$	$(p \wedge r) \vee ((p \vee s) \wedge (p \vee a))$	$p \wedge ((p \vee \sim q) \vee (\sim (\sim (z \vee \sim q))))$

Jason needs to project the cheat sheet while you solve this exercise. If he doesn't, berate him appropriately

Solution to 1

$$(a_1 \vee a_1) \wedge (a_2 \vee a_2) \wedge \cdots \wedge (a_{100} \vee a_{100}) \wedge (\sim a_1 \vee\sim a_1) \wedge (\sim a_2 \vee \sim a_2) \wedge \cdots \wedge (\sim a_{100} \vee \sim a_{100})$$

$\equiv a_1 \wedge a_2 \wedge \cdots \wedge (a_{100}) \wedge (\sim a_1) \wedge (\sim a_2) \wedge \cdots \wedge (\sim a_{100})$ *(Idempotence 100 times)*

$\equiv a_1 \wedge (\sim a_1) \wedge\ a_2 \wedge (\sim a_2) \wedge \cdots \wedge (a_{999}) \wedge (\sim a_{999}) \ldots \wedge (a_{100}) \wedge (\sim a_{100})$ *(Commutativity 100 times)*

$\equiv F \wedge F \wedge \cdots \wedge F \ldots \wedge F$ *(Negation 100 times)*

$\equiv F$ *(Idempotence 99 times)*

Solution to 2

$$(p \wedge r) \vee \big((p \vee s) \wedge (p \vee a)\big)$$

$$\equiv (p \wedge r) \vee (p \vee (s \wedge a)) \qquad \textit{(Distributivity)}$$

$$\equiv ((p \wedge r) \vee p) \vee (s \wedge a) \qquad \textit{(Associativity)}$$

$$\equiv (p \vee (p \wedge r)) \vee (s \wedge a) \qquad \textit{(Commutativity)}$$

$$\equiv p \vee (s \wedge a) \qquad \textit{(Absorption)}$$

Solution to 3

$$p \wedge \left((p \vee \sim q) \vee (\sim (\sim (z \vee \sim q))) \right)$$

$$\equiv p \wedge \left((p \vee \sim q) \vee (z \vee \sim q) \right)$$ *(Double Negation)*

$$\equiv p \wedge \left((p \vee z) \vee (\sim q \vee \sim q) \right)$$ *(Associativity)*

$$\equiv p \wedge \left((p \vee z) \vee \sim q \right)$$ *(Idempotence)*

$$\equiv p \wedge (p \vee (z \vee \sim q))$$ *(Associativity)*

$$\equiv p$$ *(Absorption)*

Boolean satisfiability problem

- At the core of computer science lies a pesky little nugget, and that nugget is SAT.
- Simplified problem statement:

Given a Boolean formula $F(x_1, x_2, \ldots, x_n)$ over n propositional symbols x_i, is there a truth assignment to the x_i that makes F true?

Well… is there?

1. $x_1 \wedge x_2 \wedge \cdots \wedge x_n$

Well… is there?

1. $x_1 \wedge x_2 \wedge \cdots \wedge x_n$ Y
2. $x_1 \wedge (\sim x_2) \wedge (x_3) \wedge \cdots \wedge (\sim x_n)$ *(if n is even, otherwise x_n)*

Well… is there?

1. $x_1 \wedge x_2 \wedge \cdots \wedge x_n$ Y
2. $x_1 \wedge (\sim x_2) \wedge (x_3) \wedge \cdots \wedge (\sim x_n)$ *(if n is even, otherwise x_n)* Y
3. $x_1 \wedge (\sim x_2) \wedge (x_3) \wedge \cdots \wedge (\sim x_1)$

Well… is there?

1. $x_1 \wedge x_2 \wedge \cdots \wedge x_n$ *Y*
2. $x_1 \wedge (\sim x_2) \wedge (x_3) \wedge \cdots \wedge (\sim x_n)$ *(if n is even, otherwise x_n)* *Y*
3. $x_1 \wedge (\sim x_2) \wedge (x_3) \wedge \cdots \wedge (\sim x_1)$ *N*
4. $x_1 \vee x_2 \vee \cdots \vee x_n$

Well… is there?

1. $x_1 \wedge x_2 \wedge \cdots \wedge x_n$ Y
2. $x_1 \wedge (\sim x_2) \wedge (x_3) \wedge \cdots \wedge (\sim x_n)$ *(if n is even, otherwise x_n)* Y
3. $x_1 \wedge (\sim x_2) \wedge (x_3) \wedge \cdots \wedge (\sim x_1)$ N
4. $x_1 \vee x_2 \vee \cdots \vee x_n$ Y
5. $x_1 \Rightarrow (x_2 \Rightarrow (\ldots (x_n \Rightarrow (\sim x_1)) \ldots))$

Well… is there?

1. $x_1 \wedge x_2 \wedge \cdots \wedge x_n$ *Y*
2. $x_1 \wedge (\sim x_2) \wedge (x_3) \wedge \cdots \wedge (\sim x_n)$ *(if n is even, otherwise x_n)* *Y*
3. $x_1 \wedge (\sim x_2) \wedge (x_3) \wedge \cdots \wedge (\sim x_1)$ *N*
4. $x_1 \vee x_2 \vee \cdots \vee x_n$ *Y*
5. $x_1 \Rightarrow (x_2 \Rightarrow (\ldots (x_n \Rightarrow (\sim x_1)) \ldots))$ *Y*

Well… is there?

1. $x_1 \wedge x_2 \wedge \cdots \wedge x_n$ *Y*
2. $x_1 \wedge (\sim x_2) \wedge (x_3) \wedge \cdots \wedge (\sim x_n)$ *(if n is even, otherwise x_n)* *Y*
3. $x_1 \wedge (\sim x_2) \wedge (x_3) \wedge \cdots \wedge (\sim x_1)$ *N*
4. $x_1 \vee x_2 \vee \cdots \vee x_n$ *Y*
5. $x_1 \Rightarrow (x_2 \Rightarrow (\ldots (x_n \Rightarrow (\sim x_1)) \ldots))$ *Y*

How easy was it to determine the last one? Did you have to mentally build a truth table?

The problem

- We can always find Boolean formulae for which we might need to calculate **an entire truth table** to find satisfiability! 
- There is **no known** algorithm that can solve satisfiability reliably, in all instances, in time that is a $poly(n)$.
  - So we can always find Boolean formulae that need exponential time to find the satisfiability of ☹
- Efficient heuristic algorithms that work well in many cases:
  - WalkSAT
  - MaxWalkSAT

SAT and P vs NP

- Most people believe that there is no algorithm that is guaranteed to solve every instance of SAT in time $poly(n)$.
- **Polynomial reduction:** A process that takes a problem and its input size n, and transforms it into SAT in time $poly(n)$.
- If a problem can be reduced to SAT in polynomial time, then it is widely speculated that there is no way to solve every instance of it in time $poly(n)$.
  - Those are called **NP-Hard problems**.

SAT and P vs NP

- If we found a polynomial algorithm for SAT, that would imply that every NP-Hard problem (so, the "hardest" in the class NP) could be solved in polynomial time.
- So P would be NP
  - Major computational implications re: cryptography, search….
- More reading:
  - NP – Hardness on Wikipedia
  - Cormen, Leiserson, Rivest, Stein, Introduction to Algorithms, Chapter 34
  - Kleinberg & Tardos, Algorithm Design, Chapter 8