administrivia

Course web: http://www.cs.washington.edu/311

Office hours: 8 office hours (by end of week)
Me: MW 2:30-3:30pm or by appointment

Call me: James or Professor James or Professor Lee

Don't: Actually call me.

Homework #1: Posted this Friday, due next Friday before class (April 10th)
Gradescope! (stay tuned)

Extra credit: Not required to get a 4.0.
Counts separately.
In total, may raise grade by ~0.1

Don't be shy (raise your hand in the back)!
Do space out your participation.

If you are not CSE yet, please do well!

logical connectives

p	$\neg p$
T	F
F	T

NOT

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

AND

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

OR

p	q	$p \oplus q$
T	T	F
T	F	T
F	T	T
F	F	F

XOR

$p \rightarrow q$

- "If p, then q" is a **promise**:
  - Whenever p is true, then q is true
  - Ask "has the promise been broken"

p	q	$p \rightarrow q$
F	F	T
F	T	T
T	F	F
T	T	T

If it's raining, then I have my umbrella.

related implications

- Implication: $p \rightarrow q$
- Converse: $q \rightarrow p$
- Contrapositive: $\neg q \rightarrow \neg p$
- Inverse: $\neg p \rightarrow \neg q$

How do these relate to each other?
How to see this?

$p \leftrightarrow q$

- p iff q
- p is equivalent to q
- p implies q and q implies p

p	q	$p \leftrightarrow q$

Roger's second sentence with a truth table

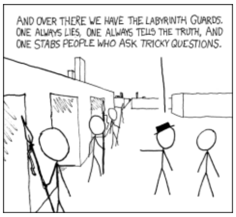
p	q	r	$q \oplus r$	$\neg q$	$((q \oplus r) \rightarrow \neg q)$	$p \rightarrow ((q \oplus r) \rightarrow \neg q)$	$(p \rightarrow ((q \oplus r) \rightarrow \neg q)) \wedge p$
T	T	T					
T	T	F					
T	F	T					
T	F	F					
F	T	T					
F	T	F					
F	F	T					
F	F	F					

Roger is only orange if whenever he either has tusks or toenails, he doesn't have tusks and he is an orange elephant."

cse 311: foundations of computing

Fall 2014

Lecture 2: Digital circuits & more logic



digital circuits

Computing with logic

- **T** corresponds to 1 or "high" voltage
- **F** corresponds to 0 or "low" voltage

Gates:

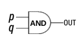
- Take inputs and produce outputs (functions)
- Several kinds of gates
- Correspond to propositional connectives

AND gate

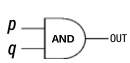
AND Connective vs. AND Gate

$p \wedge q$

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



p	q	OUT
1	1	1
1	0	0
0	1	0
0	0	0



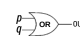
"block looks like D of AND"

OR gate

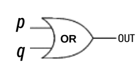
OR Connective vs. OR Gate

$p \vee q$

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



p	q	OUT
1	1	1
1	0	1
0	1	1
0	0	0



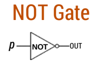
"arrowhead block looks like ∨"

NOT gate

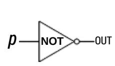
NOT Connective vs. NOT Gate (Also called *inverter*)

$\neg p$

p	$\neg p$
T	F
F	T



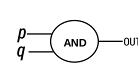
p	OUT
1	0
0	1

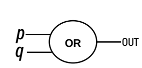


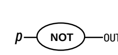
blobs are okay

"gee, thanks."

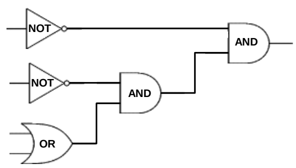
You can write gates using blobs instead of shapes.





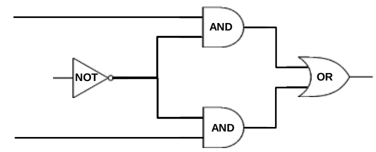


combinational logic circuits



Values get sent along wires connecting gates

combinational logic circuits



Wires can send one value to multiple gates!

logical equivalence

Terminology: A compound proposition is a…

- *Tautology* if it is always true
- *Contradiction* if it is always false
- *Contingency* if it can be either true or false

Classify!

$p \vee \neg p$

$p \oplus p$

$(p \rightarrow q) \wedge p$

$(p \wedge q) \vee (p \wedge \neg q) \vee (\neg p \wedge q) \vee (\neg p \wedge \neg q)$

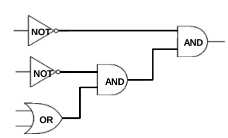
logical equivalence

Terminology: A compound proposition is a…

- *Tautology* if it is always true
- *Contradiction* if it is always false
- *Contingency* if it can be either true or false

Classify!

$$\big((p \wedge q \wedge r) \vee (\neg p \wedge q \wedge \neg r)\big) \wedge \big((p \vee q \vee \neg s) \vee (p \wedge q \wedge s)\big)$$



logical equivalence

A and B are *logically equivalent* if and only if
$A \leftrightarrow B$ is a tautology

i.e. A and B have the same truth table

The notation $A \equiv B$ denotes A and B are logically equivalent.

Example: $p \equiv \neg\neg p$

p	$\neg p$	$\neg\neg p$	$p \leftrightarrow \neg\neg p$

$A \leftrightarrow B$ vs. $A \equiv B$

$A \equiv B$ says that ***two*** propositions A and B *always* ***mean*** *the same thing.*

$A \leftrightarrow B$ is a ***single*** proposition that may be true or false depending on the truth values of the variables in A and B.

but $A \equiv B$ and $(A \leftrightarrow B) \equiv$ **T** have the same meaning.

Note: Why write $A \equiv B$ and not $A=B$?
[We use $A=B$ to say that A and B are precisely the same proposition (same sequence of symbols)]

de Morgan's laws

My code compiles or there is a bug.
[let's negate it]

Write NAND using NOT and OR:

"Always wear breathable fabrics when you get your picture taken."

de Morgan's laws

Verify: $\neg(p \wedge q) \equiv (\neg p \vee \neg q)$

p	q	$\neg p$	$\neg q$	$\neg p \vee \neg q$	$p \wedge q$	$\neg(p \wedge q)$	$\neg(p \wedge q) \leftrightarrow (\neg p \vee \neg q)$
T	T						
T	F						
F	T						
F	F						

de Morgan's laws

$\neg(p \quad q) \qquad p \qquad q$

$\neg(p \quad q) \qquad p \qquad q$

```
if !(front != null && value > front.data)
    front = new ListNode(value, front);
else {
    ListNode current = front;
    while !(current.next == null || current.next.data >= value)
        current = current.next;
    current.next = new ListNode(value, current.next);
}
```

law of implication

$(p \rightarrow q) \equiv (\neg p \vee q)$

p	q	$p \rightarrow q$	$\neg p$	$\neg p \vee q$	$(p \rightarrow q) \leftrightarrow (\neg p \vee q)$
T	T				
T	F				
F	T				
F	F				

computing equivalence

Describe an algorithm for computing if two logical expressions/circuits are equivalent.

What is the run time of the algorithm?

some familiar properties of arithmetic

- $x + y = y + x$ (commutativity)
- $x \cdot (y + z) = x \cdot y + x \cdot z$ (distributivity)
- $(x + y) + z = x + (y + z)$ (associativity)

Logic has similar algebraic properties

some familiar properties of arithmetic

- $x + y = y + x$ (commutativity)
  - $p \vee q \equiv q \vee p$
  - $p \wedge q \equiv q \wedge p$
- $x \cdot (y + z) = x \cdot y + x \cdot z$ (distributivity)
  - $p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$
  - $p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$
- $(x + y) + z = x + (y + z)$ (associativity)
  - $(p \vee q) \vee r \equiv p \vee (q \vee r)$
  - $(p \wedge q) \wedge r \equiv p \wedge (q \wedge r)$

properties of logical connectives

You will always get this list.

- Identity
  - $p \wedge \mathrm{T} \equiv p$
  - $p \vee \mathrm{F} \equiv p$
- Domination
  - $p \vee \mathrm{T} \equiv \mathrm{T}$
  - $p \wedge \mathrm{F} \equiv \mathrm{F}$
- Idempotent
  - $p \vee p \equiv p$
  - $p \wedge p \equiv p$
- Commutative
  - $p \vee q \equiv q \vee p$
  - $p \wedge q \equiv q \wedge p$
- Associative
  $(p \vee q) \vee r \equiv p \vee (q \vee r)$
  $(p \wedge q) \wedge r \equiv p \wedge (q \wedge r)$
- Distributive
  $p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$
  $p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$
- Absorption
  $p \vee (p \wedge q) \equiv p$
  $p \wedge (p \vee q) \equiv p$
- Negation
  $p \vee \neg p \equiv \mathrm{T}$
  $p \wedge \neg p \equiv \mathrm{F}$