CosC 290: Discrete Structures

Propositional Logic II

01/29/2026

Prof. Forrest

Warm up & Goals

Warm up

1. Talk to your neighbor about the best place to eat on campus

2. Let $p :=$ "it is cold outside"
   $q :=$ "it is cloudy"
   $r :=$ "Bob is happy"

   Using a truth table, determine when $(p \vee q) \rightarrow \neg r$ is false

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

Logistics

- Codelet 1 is due Friday
- Lab 1 is due Friday
- Codelet 2 is posted on the website due next Friday
- OH are Rs 10:30–1:30
  - feel free to stop by to work on things in this class or others
  - Also grab snacks
  - Also there's a fair # of students in this class want to help foster a sense of community here at in CS dept

Working through Conditionals ($p \to q$)

Contract thinking

if you get 100% on the final, then you will get an A.

P

Q

$p \to q$

if you are 3ft tall, then you can ride this ride

P → Q

Broader than NL

if I have a dog, then $2+3=5$

P → Q

p	q	p → q
T	T	T
T	F	F
F	T	T
F	F	T

if I have a dog, then $2+3=7$

Logical Equivalence

always true

Tautology : $p \vee \neg p$

Contradiction : $p \wedge \neg p$

contigency : $p \wedge q$

always false

sometimes true, sometimes false

DeMorgan's Laws

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

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

$\equiv$ means equivalent to

How do we show the validity of De Morgan's Laws?

p	$\neg p$	$p \wedge \neg p$
T	F	F
F	T	F

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

the same ⇒ they are equivalent

Verify the distributive law for propositional logic:

$(p \vee q) \vee r \equiv p \vee (q \vee r)$

Verify the distributive law for propositional logic:

$$(p \vee q) \vee r \equiv p \vee (q \vee r)$$

$n = 3$

$2^n = 8$

p	q	r	$p \vee q$	$(p \vee q) \vee r$	$q \vee r$	$p \vee (q \vee r)$
T	T	T	T	T	T	T

★ Notecard ★

Satisfiability

φ is satisfiable ⊨

Definition: A truth assignment I **satisfies** a formula φ if $\varphi[I] = T$; we write this as $I \models \varphi$. A formula φ is **satisfiable** if there exists a truth assignment I s.t. $I \models \varphi$; otherwise φ is unsatisfiable.

Definition: A formula φ is **valid** (or a tautology) if for all truth assignments I, $I \models \varphi$.

$\varphi := P \wedge \neg P$

Satisfiability Basic Examples

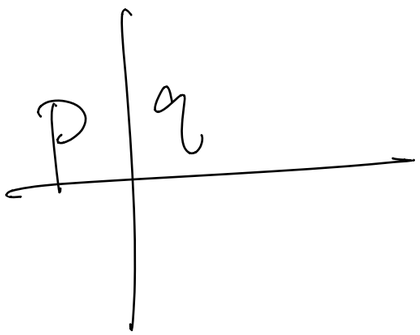
$P \wedge Q$

$P \wedge \neg P$

$P \vee \neg P$

Is the following formula valid, satisfiable, or unsatisfiable?

$$(P \rightarrow Q) \vee (Q \rightarrow P)$$



★ Notecard ★

Satisfiability Applications

Slot 1: Discrete, Data Structures

Slot 2: Privacy, Gadgets & Gizmos

$$\varphi := (P \vee Q) \wedge (R \vee S) \wedge$$

{ enrolled in something from Slot 1 and Slot 2

$$(\neg P \vee \neg Q) \wedge (\neg R \vee \neg S)$$

$$\wedge (\neg Q \vee \neg R)$$

$P :=$ enrolled in 290

$Q :=$ enrolled in 202

$R :=$ enrolled in Privacy

$S :=$ enrolled Gadgets

is φ Satisfiable?

$P = T$, $Q = T$, $R = F$, $S = T$

$P = T$, $Q = F$, $R = F$, $S = T$

Program Verification

```
def f(x: int) → int:
    if x == 0:
        y = 1
    else:
        y = 2
    return y

def main():
    . . . .
    . . . .
    # assume x is some integer (pre condition)
    y = f(x)
    assert y > 0                (post condition)
```

Will this program crash?

Propositions

$p := x = 0$

$q := y = 1$

$r := y = 2$

$c := y > 0$

φ

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

How do we use Satisfiability and φ and the post condition c to determine whether the program will crash