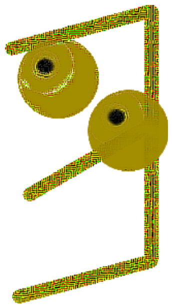
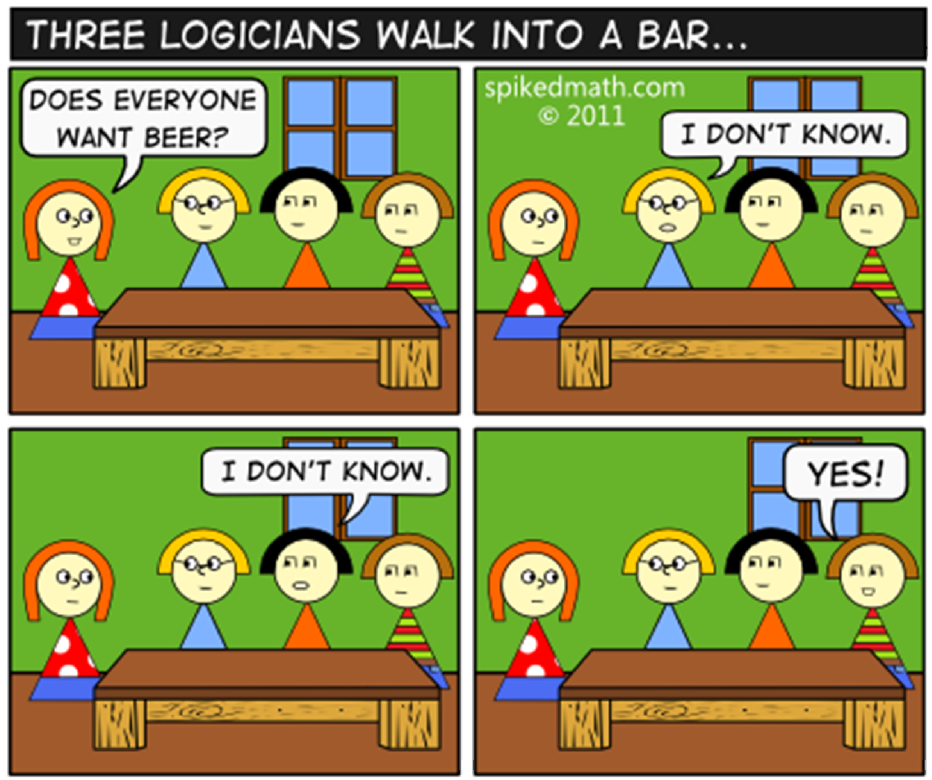
CSE 311: Foundations of Computing

Lecture 6: More Predicate Logic + Inference





Pick up HW 1 solution at front of room.

Administrative

- Homework 2 is now posted
  - Make sure your submissions are readable!

- Monday: Martin Luther King Day holiday
- Tuesday: Extra office hours
  - Jason 10:30-11:20   CSE 220
  - Josh 12:00-12:50 CSE 220

- Wednesday: Richard Anderson will teach class
  - Normal Wednesday office hours too after class.

(I will be away at workshop on complexity of proofs).

Last class: Predicates

Predicate

– **A function that returns a truth value, e.g.,**

Cat(x) ::= "x is a cat"

Prime(x) ::= "x is prime"

HasTaken(x, y) ::= "student x has taken course y"

LessThan(x, y) ::= "x < y"

Sum(x, y, z) ::= "x + y = z"

GreaterThan5(x) ::= "x > 5"

HasNChars(s, n) ::= "string s has length n"

File is Locked proposition predicate with no arguments

Predicates can have varying numbers of arguments and input types.

Last class: Domain of Discourse

For ease of use, we define one “type”/“domain” that we work over. This set of objects is called the “**domain of discourse**”.

For each of the following, what might the domain be?

(1) “x is a cat”, “x barks”, “x ruined my couch”

“mammals” or “sentient beings” or “cats and dogs” or …

(2) “x is prime”, “x = 0”, “x < 0”, “x is a power of two”

“numbers” or “integers” or “integers greater than 5” or …

(3) “student x has taken course y” “x is a pre-req for z”

“students and courses” or “university entities” or …

Last Class: Quantifiers

We use *quantifiers* to talk about collections of objects.

$\forall x\, P(x)$

$P(x)$ **is true for every** x **in the domain**

read as "for all x, P of x"

$\exists x\, P(x)$

There is an x **in the domain for which** $P(x)$ **is true**

read as "there exists x, P of x"

Last class: Statements with Quantifiers

Domain of Discourse
Positive Integers

Predicate Definitions	
Even(x) ::= "x is even"	Greater(x, y) ::= "x > y"
Odd(x) ::= "x is odd"	Equal(x, y) ::= "x = y"
Prime(x) ::= "x is prime"	Sum(x, y, z) ::= "x + y = z"

Determine the truth values of each of these statements:

$\exists$x Even(x)	**T**	**e.g. 2, 4, 6, ...**
$\forall$x Odd(x)	**F**	**e.g. 2, 4, 6, ...**
$\forall$x (Even(x) $\vee$ Odd(x))	**T**	**every integer is either even or odd**
$\exists$x (Even(x) $\wedge$ Odd(x))	**F**	**no integer is both even and odd**
$\forall$x Greater(x+1, x)	**T**	**adding 1 makes a bigger number**
$\exists$x (Even(x) $\wedge$ Prime(x))	**T**	**Even(2) is true and Prime(2) is true**

Last class: Statements with Quantifiers

Domain of Discourse
Positive Integers

Predicate Definitions	
Even(x) ::= "x is even"	Greater(x, y) ::= "x > y"
Odd(x) ::= "x is odd"	Equal(x, y) ::= "x = y"
Prime(x) ::= "x is prime"	Sum(x, y, z) ::= "x + y = z"

Translate the following statements to English

$\forall x\, \exists y$ Greater(y, x)

There is no greatest integer.

$\forall x\, \exists y$ Greater(x, y)

There is no least integer.

$\forall x\, \exists y$ (Greater(y, x) $\wedge$ Prime(y))

For every positive integer there is a larger number that is prime.

$\forall x$ (Prime(x) $\rightarrow$ (Equal(x, 2) $\vee$ Odd(x)))

Every prime number is either 2 or odd.

$\exists x\, \exists y$ (Sum(x, 2, y) $\wedge$ Prime(x) $\wedge$ Prime(y))

There exist prime numbers that differ by two."

English to Predicate Logic

Domain of Discourse
Mammals

Predicate Definitions
Cat(x) ::= "x is a cat"
Red(x) ::= "x is red"
LikesTofu(x) ::= "x likes tofu"

"Red cats like tofu"

$$\forall x\ ((Red(x) \land Cat(x)) \rightarrow LikesTofu(x))$$

"Some red cats don't like tofu"

$$\exists x\ ((Cat(x) \land Red(x)) \land \neg LikesTofu(x))$$

English to Predicate Logic

Domain of Discourse
Mammals

Predicate Definitions
Cat(x) ::= "x is a cat"
Red(x) ::= "x is red"
LikesTofu(x) ::= "x likes tofu"

"Red cats like tofu"

$$\forall x\ ((\mathrm{Red}(x) \wedge \mathrm{Cat}(x)) \rightarrow \mathrm{LikesTofu}(x))$$

"Some red cats don't like tofu"

$$\exists y\ ((\mathrm{Red}(y) \wedge \mathrm{Cat}(y)) \wedge \neg \mathrm{LikesTofu}(y))$$

English to Predicate Logic

Domain of Discourse
Mammals

Predicate Definitions
Cat(x) ::= "x is a cat"
Red(x) ::= "x is red"
LikesTofu(x) ::= "x likes tofu"

When putting two predicates together like this, we use an "and".

"Red cats like tofu"

When restricting to a smaller domain in a "for all" we use implication.

When there's no leading quantification, it means "for all".

"Some red cats don't like tofu"

When restricting to a smaller domain in an "exists" we use and.

"Some" means "there exists".

Negations of Quantifiers

Predicate Definitions
PurpleFruit(x) ::= “x is a purple fruit”

(*) $\forall x$ PurpleFruit(x) (“All fruits are purple”)

What is the negation of (*)?

(a) “there exists a purple fruit”

(b) “there exists a non-purple fruit”

(c) “all fruits are not purple”

Try your intuition! Which one “feels” right?

Key Idea: In every domain, exactly one of a statement and its negation should be true.

Negations of Quantifiers

Predicate Definitions
PurpleFruit(x) ::= "x is a purple fruit"

(*) $\forall x$ PurpleFruit(x) ("All fruits are purple")

What is the negation of (*)?

~~(a)~~ "there exists a purple fruit"

(b) "there exists a non-purple fruit"

~~(c) "all fruits are not purple"~~

Key Idea: In every domain, exactly one of a statement and its negation should be true.

Domain of Discourse
{plum}

(*), (a)

Domain of Discourse
{apple}

(b), (c)

Domain of Discourse
{plum, apple}

(a), (b)

Negations of Quantifiers

Predicate Definitions
PurpleFruit(x) ::= "x is a purple fruit"

(*) $\forall x$ PurpleFruit(x) ("All fruits are purple")

What is the negation of (*)?

(a) "there exists a purple fruit"

(b) "there exists a non-purple fruit"

(c) "all fruits are not purple"

Key Idea: In every domain, exactly one of a statement and its negation should be true.

Domain of Discourse
{plum}

(*), (a)

Domain of Discourse
{apple}

(b), (c)

Domain of Discourse
{plum, apple}

(a), (b)

The only choice that ensures exactly one of the statement and its negation is (b).

De Morgan's Laws for Quantifiers

$$\boxed{\begin{array}{c} \neg \forall x\, P(x) \equiv \exists x\, \neg P(x) \\ \neg \exists x\, P(x) \equiv \forall x\, \neg P(x) \end{array}}$$

$\exists \leftrightarrow \vee$

$\forall \leftrightarrow \wedge$

De Morgan's Laws for Quantifiers

$$\neg \forall x\, P(x) \equiv \exists x\, \neg P(x)$$
$$\neg \exists x\, P(x) \equiv \forall x\, \neg P(x)$$

There is a largest integer

"There is no largest integer"

GreaterThan(x, y) or EqualTo

$$\neg \exists x\, \forall y\ (x \geq y)$$
$$\equiv\ \forall x\, \neg \forall y\ (x \geq y) \qquad \text{De Morgan}$$
$$\equiv\ \forall x\ \exists y\, \neg (x \geq y) \qquad \text{De Morgan}$$
$$\equiv\ \forall x\ \exists y\ (y > x)$$

"For every integer there is a larger integer"

De Morgan's Laws for Quantifiers

$$\neg \forall x\, P(x) \equiv \exists x\, \neg P(x)$$
$$\neg \exists x\, P(x) \equiv \forall x\, \neg P(x)$$

Negation of "Red cats like tofu"

$\neg \forall x\, ((\mathrm{Red}(x) \wedge \mathrm{Cat}(x)) \rightarrow \mathrm{LikesTofu}(x))$

$\equiv \exists x\, \neg((\mathrm{Red}(x) \wedge \mathrm{Cat}(x)) \rightarrow \mathrm{LikesTofu}(x))$ De Morgan

$\equiv \exists x\, \neg(\neg(\mathrm{Red}(x) \wedge \mathrm{Cat}(x)) \vee \mathrm{LikesTofu}(x))$ Implication

$\equiv \exists x\, (\neg \neg(\mathrm{Red}(x) \wedge \mathrm{Cat}(x)) \wedge \neg \mathrm{LikesTofu}(x))$ De Morgan

$\equiv \exists x\, ((\mathrm{Red}(x) \wedge \mathrm{Cat}(x)) \wedge \neg \mathrm{LikesTofu}(x))$ Double Neg

"Some red cats don't like tofu."

Scope of Quantifiers

$\exists x\ (P(x) \wedge Q(x))$ **vs.** $\exists x\, P(x) \wedge \exists x\, Q(x)$

Domain	P	Q
1	F	F
2	F	T
3	T	F
4	F	F
5	T	F
⋮		
	T	T

scope of quantifiers

$\exists x \; (P(x) \wedge Q(x))$ **vs.** $\exists x \; P(x) \wedge \exists x \; Q(x)$

This one asserts P and Q of the *same* x.

This one asserts P and Q of potentially different x's.

Scope of Quantifiers

Example: $\text{NotLargest}(x) \equiv \exists y\ \text{Greater}(y, x)$
$\equiv \exists z\ \text{Greater}(z, x)$

truth value:

doesn't depend on y **or** z **"bound variables"**

does depend on x **"free variable"**

quantifiers only act on free variables of the formula they quantify

$$\forall x\ (\exists y\ (P(x,y) \rightarrow \forall x\ Q(y, x)))$$

Quantifier "Style"

$$\forall x(\exists y\ (P(x,y) \rightarrow \forall\ x\ Q(y, x)))$$

This isn't "wrong", it's just horrible style.
Don't confuse your reader by using the same variable multiple times…there are a lot of letters…

Nested Quantifiers

- **Bound variable names don't matter**

$$\forall x\ \exists y\ P(x, y) \equiv \forall a\ \exists b\ P(a, b)$$

- **Positions of quantifiers can sometimes change**

$$\forall x\ (Q(x) \land \exists y\ P(x, y)) \equiv \forall x\ \exists y\ (Q(x) \land P(x, y))$$

- **But: order is important...**

Quantifier Order Can Matter

Domain of Discourse
Integers OR {1, 2, 3, 4}

Predicate Definitions
GreaterEq(x, y) ::= "x ≥ y"

x \ y	1	2	3	4
1	T	F	F	F
2	T	T	F	F
3	T	T	T	F
4	T	T	T	T

"There is a number greater than or equal to all numbers."

$$\exists x\, \forall y\, \text{GreaterEq}(x, y)$$

"Every number has a number greater than or equal to it."

$$\forall y\, \exists x\, \text{GreaterEq}(x, y)$$

← true for both

The purple statement requires an entire row to be true.
The red statement requires one entry in each column to be true.

False for integers

Integers

Quantification with Two Variables

expression	when **true**	when **false**
$\forall x \, \forall y \, P(x, y)$ $\equiv \forall y \, \forall x \, P(x,y)$	Every pair is true.	At least one pair is false.
$\exists x \, \exists y \, P(x, y)$ $\equiv \exists y \, \exists x \, P(x,y)$	At least one pair is true.	All pairs are false.
$\forall x \, \exists y \, P(x, y)$ not $\not\equiv$	We can find a specific y for each x. $(x_1, y_1), (x_2, y_2), (x_3, y_3)$	Some x doesn't have a corresponding y.
$\exists y \, \forall x \, P(x, y)$	We can find ONE y that works no matter what x is. $(x_1, y), (x_2, y), (x_3, y)$	For any candidate y, there is an x that it doesn't work for.

Logical Inference

Logical Inference

- So far we've considered:
  - How to understand and *express* things using propositional and predicate logic
  - How to *compute* using Boolean (propositional) logic
  - How to show that different ways of expressing or computing them are *equivalent* to each other

- Logic also has methods that let us *infer* implied properties from ones that we know
  - Equivalence is a small part of this

Applications of Logical Inference

- Software Engineering
  - Express desired properties of program as set of logical constraints
  - Use inference rules to show that program implies that those constraints are satisfied
- Artificial Intelligence
  - Automated reasoning
- Algorithm design and analysis
  - e.g., Correctness, Loop invariants.
- Logic Programming, e.g. Prolog , SAT solvers
  - Express desired outcome as set of constraints
  - Automatically apply logic inference to derive solution

Proofs

- Start with hypotheses and facts
- Use rules of inference to extend set of facts
- Result is proved when it is included in the set

An inference rule: *Modus Ponens*

- If p and p $\rightarrow$ q are both true then q must be true

- Write this rule as $$\frac{p,\ p \rightarrow q}{\therefore\ q}$$

- Given:
  - If it is Monday then you have a 311 class today.
  - It is Monday.

- Therefore, by Modus Ponens:
  - You have a 311 class today.

My First Proof!

$$\frac{p,\ p \to q}{\therefore q}$$

Show that r follows from p, $p \to q$, and $q \to r$

1. p Given
2. $p \to q$ Given
3. $q \to r$ Given
4. q By Modus Ponens from 1 and 2
5. r By M.P from 4 and 3

My First Proof!

$p, p \to q$

$\therefore q$

Show that r follows from p, p → q, and q → r

1.	p	Given
2.	$p \to q$	Given
3.	$q \to r$	Given
4.	q	MP: 1, 2
5.	r	MP: 3, 4

Proofs can use equivalences too

$$\frac{p,\ p \to q}{\therefore q}$$

Show that $\neg p$ follows from $p \to q$ and $\neg q$

1.	$p \to q$	Given
2.	$\neg q$	Given
3.	$\neg q \to \neg p$	Contrapositive: 1
4.	$\neg p$	MP: 2, 3

Inference Rules

- Each inference rule is written as:
  ...which means that if both A and B are true then you can infer C and you can infer D.

$$\frac{A,\ B}{\therefore\ C, D}$$

  – For rule to be correct $(A \wedge B) \rightarrow C$ and $(A \wedge B) \rightarrow D$ must be a tautologies

- Sometimes rules don't need anything to start with. These rules are called axioms:
  – e.g. *Excluded Middle Axiom*

$$\frac{\quad}{\therefore\ p \vee \neg p}$$

$$\frac{\quad}{\therefore\ p \rightarrow p}$$