CSE 311 Foundations of Computing I

Lecture 5

Predicate Logic

Spring 2013

Announcements

- Reading assignments
  - Predicates and Quantifiers
    - 1.4, 1.5 7th Edition
    - 1.3, 1.4 6th Edition
- Hand in Homework 1 now
  - Homework 2 is available on the website

Highlights from Last Lecture

Sum-of-products canonical form

- Also known as Disjunctive Normal Form (DNF)
- Also known as minterm expansion

F = *001* *011* *101* *110* *111*

F = A′B′C + A′BC + AB′C + ABC′ + ABC

A	B	C	F	F′
0	0	0	0	1
0	0	1	1	0
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	0

F′ = A′B′C′ + A′BC′ + AB′C′

Highlights from Last Lecture

Product-of-sums canonical form

- Also known as Conjunctive Normal Form (CNF)
- Also known as maxterm expansion

F = *000* *010* *100*

F = (A + B + C) (A + B′ + C) (A′ + B + C)

A	B	C	F	F′
0	0	0	0	1
0	0	1	1	0
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	0

F′ = (A + B + C′) (A + B′ + C′) (A′ + B + C′) (A′ + B′ + C) (A′ + B′ + C′)

Highlights from Last Lecture

S-o-P, P-o-S, and de Morgan's theorem

- Complement of function in sum-of-products form
  - $F' = A'B'C' + A'BC' + AB'C'$
- Complement again and apply de Morgan's and get the product-of-sums form
  - $(F')' = (A'B'C' + A'BC' + AB'C')'$
  - $F = (A + B + C)\ (A + B' + C)\ (A' + B + C)$

- Complement of function in product-of-sums form
  - $F' = (A + B + C')\ (A + B' + C')\ (A' + B + C')\ (A' + B' + C)\ (A' + B' + C')$
- Complement again and apply de Morgan's and get the sum-of-product form
  - $(F')' = (\ (A + B + C')(A + B' + C')(A' + B + C')(A' + B' + C)(A' + B' + C')\ )'$
  - $F = A'B'C + A'BC + AB'C + ABC' + ABC$

Predicate Logic

- *Predicate* or *Propositional Function*
  - A function that returns a truth value
- "*x* is a cat"
- "*x* is prime"
- "student *x* has taken course *y*"
- "$x > y$"
- "$x + y = z$" or Sum(x, y, z)

NOTE: We will only use predicates with variables or constants as arguments.

Quantifiers

- $\forall x\ P(x)$: $P(x)$ is true for every x in the domain
- $\exists x\ P(x)$: There is an x in the domain for which $P(x)$ is true

Statements with quantifiers

Domain:
Positive Integers

Even(*x*)
Odd(*x*)
Prime(*x*)
Greater(*x*,*y*)
Equal(*x*,*y*)

- $\exists x$ Even(x)
- $\forall x$ Odd(x)
- $\forall x$ (Even(x) $\vee$ Odd(x))
- $\exists x$ (Even(x) $\wedge$ Odd(x))
- $\forall x$ Greater(x+1, x)
- $\exists x$ (Even(x) $\wedge$ Prime(x))

Statements with quantifiers

Domain:
Positive Integers

Even(x)
Odd(x)
Prime(x)
Greater(x,y)
Equal(x,y)
Sum(x,y,z)

- $\forall x \exists y$ Greater (y, x)
- $\forall x \exists y$ Greater (x, y)
- $\forall x \exists y$ (Greater(y, x) $\wedge$ Prime(y))
- $\forall x$ (Prime(x) $\rightarrow$ (Equal(x, 2) $\vee$ Odd(x))
- $\exists x \exists y$(Sum(x, 2, y) $\wedge$ Prime(x) $\wedge$ Prime(y))

Statements with quantifiers

Domain:
Positive Integers

Even(x)
Odd(x)
Prime(x)
Greater(x,y)
Sum(x,y,z)

- "There is an odd prime"
- "If x is greater than two, x is not an even prime"
- $\forall x \forall y \forall z$ ((Sum(x, y, z) $\wedge$ Odd(x) $\wedge$ Odd(y))$\rightarrow$ Even(z))
- "There exists an odd integer that is the sum of two primes"

English to Predicate Calculus

- "Red cats like tofu"

Cat(x)
Red(x)
LikesTofu(x)

Goldbach's Conjecture

- Every even integer greater than two can be expressed as the sum of two primes

Even(x)
Odd(x)
Prime(x)
Greater(x,y)
Equal(x,y)

Domain:
Positive Integers

Scope of Quantifiers

- Notlargest(x) $\equiv \exists\, y$ Greater (y, x)
  $\equiv \exists\, z$ Greater (z, x)
  - Value doesn't depend on y or z "bound variables"
  - Value 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)))

Scope of Quantifiers

- $\exists x\ (P(x) \wedge Q(x))$ vs $\exists x\, P(x) \wedge \exists x\, Q(x)$

Nested Quantifiers

- Bound variable name doesn't matter
  - $\forall$ x $\exists$ y P(x, y) $\equiv$ $\forall$ a $\exists$ b P(a, b)
- Positions of quantifiers can change
  - $\forall$ x (Q(x) $\wedge$ $\exists$ y P(x, y)) $\equiv$ $\forall$ x $\exists$ y (Q(x) $\wedge$ P(x, y))
- BUT: Order is important...

Quantification with two variables

Expression	When true	When false
$\forall$x $\forall$ y P(x, y)		
$\exists$ x $\exists$ y P(x, y)		
$\forall$ x $\exists$ y P(x, y)		
$\exists$ y $\forall$ x P(x, y)		

Negations of Quantifiers

- Not every positive integer is prime
- Some positive integer is not prime
- Prime numbers do not exist
- Every positive integer is not prime

De Morgan's Laws for Quantifiers

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

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 no largest integer"

$$\neg \exists x \forall y\ (x \geq y)$$
$$\equiv \forall x \neg \forall y\ (x \geq y)$$
$$\equiv \forall x\ \exists y \neg (x \geq y)$$
$$\equiv \forall x\ \exists y\ (y > x)$$

"For every integer there is a larger integer"