FCS Week 2 Lecture Note

Notebook:	Fundamentals of Computer Science
Created:	2021-04-13 10:30 AM
Updated:	2021-04-14 4:52 PM
Author:	SUKHJIT MANN

Cornell Notes	**Topic:** Logic: Part 2	Course: BSc Computer Science Class: CM1025 Fundamentals of Computer Science[Lecture] Date: April 14, 2021

Essential Question:

What is propositional logic?

Questions/Cues:

- What is propositional equivalence?
- What are De Morgan's Laws?
- What are some other important equivalences?
- What is the Contrapositive?
- What are Predicates and Quantifiers?
- What is the Existential Quantifier?
- What is the Universal Quantifier?
- How do we defined quantified statements using connectives?
- How is De Morgan's Law applied to quantifiers?

Notes

Propositional Equivalences

- Formula A and B are equivalent if they have the same truth tables
- We denote the equivalence by $\equiv$
- $A \equiv B$ means that A and B always have the same truth values, regardless of how the variables are assigned.
- Note that $\equiv$ is NOT a connective

De Morgan's Laws

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

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

Example: negate the following

- It is Wednesday and it is not sunny
- NOT (It is Wednesday and it is not sunny)
- NOT (It is Wednesday) or NOT (it is not sunny)
- It is NOT Wednesday or it is sunny

Truth table for $\neg(p \wedge q) \equiv \neg p \vee \neg q$

p	q	$\neg(p \wedge q)$	$\neg p$	$\neg q$	$\neg p \vee \neg q$
1	1	0	0	0	0
1	0	1	0	1	1
0	1	1	1	0	1
0	0	1	1	1	1

Important equivalence

- $(p \to q) \equiv (\neg p \vee q)$
- Can you write the right hand side using conjunction?
- Use De Morgan's Law
- $(\neg p \vee q) \equiv \neg(\neg\neg p \wedge \neg q) \equiv \neg(p \wedge \neg q)$
- So $(p \to q) \equiv \neg(p \wedge \neg q)$

Another Equivalence: Contrapositive

- $p \to q \equiv \neg q \to \neg p$
- Why is it true?
- $p \to q \equiv \neg p \vee q$
- $\neg p \vee q \equiv q \vee \neg p \equiv \neg q \to \neg p$

$$a \to b \equiv \neg a \vee b$$

- Predicates that describe properties of objects
  - Odd(3) means 3 is an odd number; Odd is a predicate, 3 is an object
  - Equal(5,6) means 5 and 6 are equal; Equal is a predicate, 5,6
- Predicates take arguments and become propositions
- The connectives for propositional logic apply the same way
  - Odd (3) ∧ Prime (3), which means 3 is odd and 3 is prime, true.
  - Even (4) → Prime (4), which means 4 is even then 4 is prime, false.
- Quantifiers make reasoning on multiple objects
- The objects for the quantified statements are chosen from a Domain

- Existential quantifier, denoted by ∃
  - ∃x "some formula"
  - Means for some x, the statement "some formula" is true.
  - Example: ∃x Odd(x) it means some numbers are odd
  - ∃x Prime(x) it means there exists at least one number that is prime
  - NOTE, it is enough to find one element to make the formula true

- Universal quantifier, denoted by ∀
  - ∀x "some formula"
  - Means for **ALL x**, the statement "some formula" is true
  - Example: ∀x (Odd(x) ∨Even(x)) – it means all numbers are either odd or even
  - ∀x (x < x+1) – it means all numbers increase when you add 1
  - NOTE, it is **NOT enough** to find some elements that make the formula true
  - It must be true for **all elements**

Universal Quantifier

- **"All Ps are Qs" translates as ∀x (P(x) → Q(x))**
- A counter-example proves that a universally quantified statement is false
- Example: ∀x (Prime(x) → Odd(x)), which means all prime numbers are odd
- Let x be 2. But 2 is prime and not odd, so the statement is false
- ∀x (Multiple4(x) → Multiple2(x)), which means all multiples of 4 are multiples of 2. True.
- **"No Ps are Qs" Translates as ∀x (P(x) → ¬Q(x))**
- If we find one P that is Q then we prove that the statement above is false
- Example: ∀x (Prime(x) → ¬Square(x)), which means no prime number is square number. True.
- ∀x (Prime(x) → ¬Even(x)), which means no prime number is even. False.
  - Remember to prove something is true universally is much more challenging than showing by counterexample that something is false

Existential Quantifier

- **"Some Ps are Qs" translates as ∃x (P(x) ∧ Q(x))**
- Existentially quantified statements are true if an evidence example exists
- Example: ∃x (Prime(x)∧Even(x)) – some prime numbers are even, true
- Example: ∃x (Professor(x)∧ Under2 (x)) – some professors are under 2. Not true!
- **"Some Ps are not Qs" translates as ∃x (P(x) ∧ ¬ Q(x))**
- Example: ∃x (Prime(x)∧ ¬Even(x)) – some prime numbers are not even. True

Quantifiers to connectives

- $\exists x, P(x)$ and domain is D = $\{x_1, x_2, .., x_n\}$ means

$$P(x_1) \vee P(x_2) \vee \cdots \vee P(x_n)$$

- $\forall x, P(x)$ and domain is D = $\{x_1, x_2, .., x_n\}$ means

$$P(x_1) \wedge P(x_2) \wedge \cdots \wedge P(x_n)$$

De Morgan's Law

- $\exists x, P(x)$ and domain is D = $\{x_1, x_2, .., x_n\}$ means
  $P(x_1) \vee P(x_2) \vee \cdots \vee P(x_n)$

- $\neg\exists x, P(x) \equiv \neg((P(x_1) \vee P(x_2) \vee \cdots \vee P(x_n))$

  $\equiv \neg(P(x_1) \wedge \neg(P(x_2) \wedge \ldots \wedge \neg(P(x_n)$

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

De Morgan's Law

- $\forall x, P(x)$ and domain is D = $\{x_1, x_2, .., x_n\}$ means
  $P(x_1) \wedge P(x_2) \wedge \cdots \wedge P(x_n)$

- $\neg\forall x, P(x) \equiv \neg((P(x_1) \wedge P(x_2) \wedge \cdots \wedge P(x_n))$

  $\equiv \neg(P(x_1) \vee \neg(P(x_2) \vee \ldots \vee \neg(P(x_n)$

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

Negate the following

De Morgan's Law
$\neg(p \vee q) \equiv \neg p \wedge \neg q$

- $\forall x(p(x) \to q(x))$

- $\neg\forall x(p(x) \to q(x)) \equiv \exists x \neg(p(x) \to q(x))$

  $\equiv \exists x \neg(\neg p(x) \vee q(x))$

  $\equiv \exists x\big(\neg\neg p(x) \wedge \neg q(x)\big)$

Summary

In this week, we learned about logical equivalences and quantifiers.