FCS Week 1 Lecture Note

Notebook:	Fundamentals of Computer Science		
Created:	2021-04-13 10:30 AM	**Updated:**	2021-04-21 3:08 PM
Author:	SUKHJIT MANN		

Cornell Notes	**Topic:** Logic: Part 1	Course: BSc Computer Science Class: CM1025 Fundamentals of Computer Science[Lecture] Date: April 13, 2021

Essential Question:

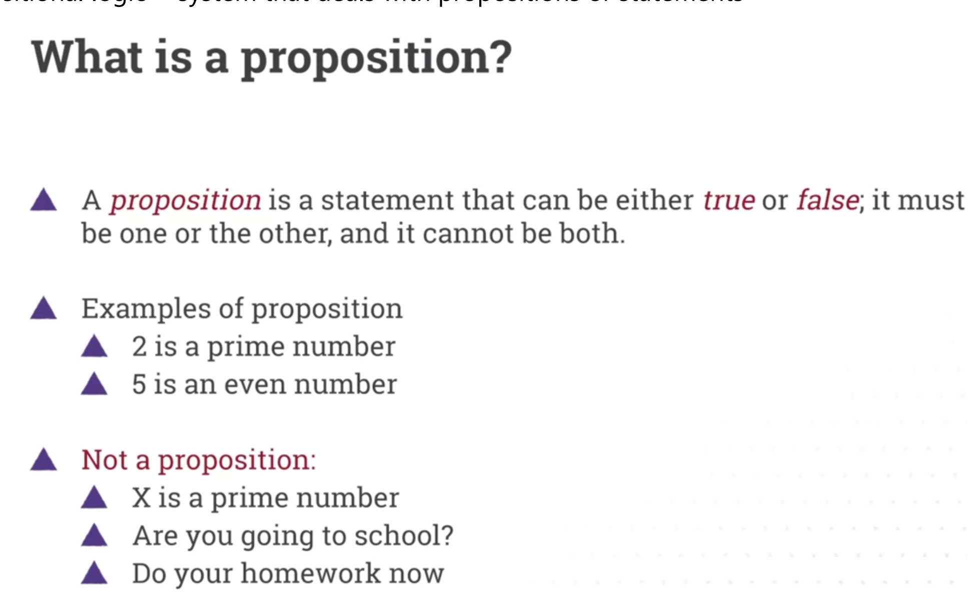
What is propositional logic?

Questions/Cues:

- What is meant by propositional logic?
- What is a proposition?
- What are basic building blocks of logic?
- What is a truth table?
- What is the order of precedence for the logical connectives in propositional logic?
- What is a Tautology?
- What is a Consistency or Contradiction?

Notes

- Propositional logic = system that deals with propositions or statements



- Logical NOT: ¬
  - ¬ p is true if and only if p is false
  - It is also called negation

- Logical OR: ∨
  - p ∨ q is true if and only if at least one of p or q are true.
  - It is also called disjunction

- Logical AND : ∧
  - p ∧ q is true if and only if both p and q are true.
  - It is also called conjunction

- Logical IF...THEN: →
  - p → q is true if and only if either p is false or q is true.
  - It is also called conditional or implication
  - p is the premise, q is the conclusion

- Logical IF and only IF: ↔
  - p ↔ q is true if and only if both p and q are true.or both are false
  - It is also called Logical bi-conditional

- Exclusive OR: XOR, ⊕
  - p ⊕ q is true if p or q is true but not both.

P = I study 20 hours a week	R = I will pass the exam
S = I will be happy	Q = I attend all the lectures

- (P ∨ Q) → (R ∧ S)
- → means if ...then
- if (P ∨ Q) then (R ∧ S)
- ∨ means or
- If (P or Q) then (R ∧ S)
- ∧ means and
- If (P or Q) then (R and S)
- If I study 20 hours a week or I attend all the lectures then I will pass the exam and I will be happy

If UK does not exit EU then skilled nurses will not leave the NHS and research grants will remain intact.

- Let's name the atomic propositions:
  - P=UK exits the EU, Q=Nurses leave NHS, S=Research grants will remain intact
- Rewrite
  - If (not P) then (not Q and S)
- Connectives:
  - If (not P) then (not Q and S)
  - not P → (not Q and S)
  - ¬P → (¬Q and S)

- Truth table = set of all possible outcomes of propositions and connectives
  - A proposition can either be true or false
  - A general proposition in contrast is where a variable signifies a hypothetical containers that holds any statement whose truth value is not specified
  - The number of rows in a truth table depends on the number of given propositions, if we have n proposition in our truth table, our truth table will have 2^n rows.

Negation: ¬

p	$\neg p$
1	0
0	1

Negation is equivalent to 1 - T(p)

Conjunction: ∧

p	q	$p \wedge q$
1	1	1
1	0	0
0	1	0
0	0	0

- Conjunction is equivalent to Multiplication of T(p) x T(q)

Disjunction: $\vee$

p	q	$p \vee q$
1	1	1
1	0	1
0	1	1
0	0	0

- Disjunction is equivalent to the Maximum: Max {T(p), T(q)}

Implication: $\rightarrow$

p	q	$p \rightarrow q$
1	1	1
1	0	0
0	1	1
0	0	1

- Implication is like making a promise.
- Example: If you study then I give you a cookie.
- The implication is only false when a promise is broken.

- Mathematically, implication can be thought of as less than or equal to.
  - When p is 1 and q is 0, then p is not less than or equal to 0, so the inequality doesn't hold. For any other truth value, the inequality holds

Bi-conditional: $\leftrightarrow$

p	q	$p \leftrightarrow q$
1	1	1
1	0	0
0	1	0
0	0	1

Bi-conditional is equivalent to equality check: T(p) = T(q)

- Bi-conditional is only true when p and q have the same truth value

Exclusive or: XOR

p	q	p XOR q
1	1	0
1	0	1
0	1	1
0	0	0

- Exclusive or is equivalent to inequality check: T(p) ≠ T(q)
- It is the opposite of the bi-conditional

- Exclusive OR means that only one of p or q is true. The difference between exclusive OR and normal disjunction is that when p and q are both true or both false, then exclusive OR outputs false

Operator precedence for propositional logic

$\neg \quad \wedge \quad \vee \quad \rightarrow \quad \leftrightarrow$

- How to parse the following: $p \rightarrow p \wedge \neg q \vee s$
- $\neg$, negation binds to what immediately follows
- $p \rightarrow p \wedge (\neg q) \vee s$
- $\wedge$, and binds its left and right neighbours
- $p \rightarrow (p \wedge (\neg q)) \vee s$
- $\vee$, or binds its left and right neighbours
- $p \rightarrow ((p \wedge (\neg q)) \vee s)$

p	q	$p \wedge q$	$(p \wedge q) \rightarrow p$
1	1	1	1
1	0	0	1
0	1	0	1
0	0	0	1

Conclusion **Premise**

- Remember, implication is false only when the premise is true and the conclusion is false

Compare $(p \wedge q) \rightarrow p$ and $p \wedge (q \rightarrow p)$

p	q	$(p \wedge q) \rightarrow p$	$p \wedge (q \rightarrow p)$
1	1	1	1
1	0	1	1
0	1	1	0
0	0	1	0

- Tautology = No matter the truth values of the propositions, the output is true, or in other words something that is always true

Tautology

- A formula that is always true
- Example: $p \vee \neg p$
  - $(p \wedge q) \rightarrow p$
  - $(p \wedge q) \rightarrow q$
- Not a tautology:
  - $p \wedge q$
  - Because if T(p) = 0 then $p \wedge q$ is false
  - $p \rightarrow q$
  - Because if T(p) = 1 and T(q) = 0 then $p \rightarrow q$ is false

- Consistency = is true for at least one scenario

Consistent

- A formula that is true at least for one scenario
- Example $p \wedge q$
- All the connectives are consistent

- Not consistent or inconsistent is a formula that is never true
- $p \wedge \neg p$
- $(p \vee \neg p) \rightarrow (p \wedge \neg p)$
- It is also called Contradiction

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

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

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

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

Consistent

- Consistent because for at least one scenario it's true

$(p \to q) \to (\neg q \vee r)$

Consistent

p	q	r	$p \to q$	$\neg q$	$(\neg q \vee r)$	$(p \to q) \to (\neg q \vee r)$
1	1	1	0	0	1	1
1	1	0	1	0	0	0
1	0	1	0	1	1	1
1	0	0	0	1	1	1
0	1	1	1	0	1	1
0	1	0	1	0	0	0
0	0	1	1	1	1	1
0	0	0	1	1	1	1

$$(p \to q) \leftrightarrow (\neg p \vee q)$$

p	q	$p \to q$	$\neg p$	$\neg p \vee q$	$(p \to q) \leftrightarrow (\neg p \vee q)$
1	1	1	0	1	1
1	0	0	0	0	1
0	1	1	1	1	1
0	0	1	1	1	1

Summary

In this week, we learned about the basic building blocks of propositional logic, their order and their use.