cse541
LOGIC FOR COMPUTER SCIENCE

Professor Anita Wasilewska

Spring 2015

LECTURE 2

Chapter 2
Introduction to Classical Propositional Logic

PART 1: Classical Propositional Model Assumptions

PART 2: Syntax and Semantics

PART 3: Classical Propositional Connectives

Chapter 2
Introduction to Classical Propositional Logic

PART 1: Classical Propositional Model Assumptions

Very short History

Origins: Stoic school of philosophy (3rd century B.C.), with the most eminent representative was Chryssipus.

Modern Origins: Mid-19th century

English mathematician G. Boole, who is sometimes regarded as the founder of mathematical logic.

First Axiomatic System: 1879 by German logician G. Frege.

Classical Propositional Model

Assumption 1

The first assumption of the propositional model of classical reasoning, and hence of a formalization of classical propositional logic is the following.

We assume that sentences (statements) are always evaluated as true or false.

Such sentences are called logical sentences or propositions

Hence the name propositional logic (two valued)

Motivation

Why two logical values only?

We build a model for classical logic to reflect the black and white qualities of mathematics.

We expect from mathematical theorems to be always either true or false and the reasonings leading to them should guarantee this without any ambiguity.

Classical Propositional Model

Assumption 2

1. We combine logical sentences (basic true- false blocks) to form more complicated sentences, called formulas

2. We combine logical sentences using only only the following words or phrases:
not; *and*; *or*; *if ..., then*; *if and only if*

3. We use **symbols** do denote both **logical sentences** and the words or phrases, called **logical connectives**

Hence the name **symbolic logic**

Choice of the Symbols

There are different choices of **logical symbols**; we adopt the following

Symbols for logical sentences are

$a, b, c, p, r, q, ...,$ with indices, if necessary

They are called **propositional variables**

Symbols for **logical connectives** are:

$\neg$ for "*not*",

$\cap$ for "*and*", $\cup$ for "*or*",

$\Rightarrow$ for *"if ..., then"*, and $\Leftrightarrow$ for *"if and only if"*.

The **names** for our **logical connectives** are:

$\neg$ negation

$\cap$ conjunction, $\cup$ disjunction,

$\Rightarrow$ implication and $\Leftrightarrow$ equivalence.

Translation Example

Exercise: Translate a natural language sentence into corresponding propositional symbolic logic **formula**.

Sentence

The fact that it is not true that at the same time 2+2 = 4 and 2+2 = 5 implies that 2+2 = 4

Translation Steps

Step 1: identify all **logical connectives** and we write the sentence introducing parenthesis to express the meaning of the sentence

If not $(2+2=4$ *and* $2+2=5)$ *then* $2+2=4$

Translation Example

Step 2: identify basic sentences with no logical connectives and assign propositional variables to them:

$$a: \quad 2+2=4, \qquad b: \quad 2+2=5$$

Step 3 : we write the (symbolic) **formula** as

$$(\neg(a \cap b) \Rightarrow a)$$

PART 2: Syntax

Syntax of a symbolic language is the formal description of the symbols we use and the way we construct its set of formulas

A formal language, or just a language, is another word for the symbolic language

Propositional languages are the syntax of propositional logics

Predicate languages form the syntax of more complex logics, called predicate logics or predicate calculi

General Remarks

The formal language symbols and well defined set of formulas i.e. an established **syntax** do not directly carry with them any logical value

We assign a **logical value** to syntactically defined formulas of a given language in a **separate step**

This next step is called a semantics of the given language

We will see that a given language can have different semantics and the different semantics will define different logics

Propositional Formulas

Propositional formulas are expressions build recursively by means of logical connectives and propositional variables as follows

1. All propositional variables are are **formulas**

They are called **atomic formulas**

2. For already defined formulas $A, B,$ the expressions

$$(A \cap B),\ (A \cup B),\ (A \Rightarrow B),\ (A \Leftrightarrow B),\ \neg A$$

are also **well defined formulas**

They are called **non-atomic formulas**

Example

By the definition, any propositional variable is a formula. Let's take two variables a and b.

By the recursive step we get that

$$(a \cap b),\ (a \cup b),\ (a \Rightarrow b),\ (a \Leftrightarrow b),\ \neg a,\ \neg b$$

are formulas

Recursive step applied again produces for example formulas :

$$\neg(a \cap b),\ \ ((a \Leftrightarrow b) \cup \neg b),\ \ \neg\neg a,\ \ \neg\neg(a \cap b)$$

Formulas

We didn't list all formulas we obtained in the first recursive step.

Moreover, the recursive process could continue.

The set of all formulas is countably infinite.

Remark that we put parenthesis within the formulas in a way to avoid ambiguity

The expression: $a \cap b \cup a$, is ambiguous.

We don't know whether it represents $(a \cap b) \cup a$ or $a \cap (b \cup a)$

Observe that neither of $a \cap b \cup a$, $(a \cap b) \cup a$ or $a \cap (b \cup a)$ is a well formed **formula**

Introduction to Semantics

We explain now how we define propositional connectives in terms of logical values and discuss the motivations for presented definitions

The **formal description** of a process of assigning logical values to all formulas of a given language is called a semantics of the language

Conjunction: Motivation and Definition

A **conjunction** $(A \cap B)$ is a true formula if both A and B are true formulas.

If one of the formulas, or both, are false, then the **conjunction** is a false formula

Let's denote statement: formula A is **false** by $A = F$ and

a statement: formula A is **true** by $A = T$

Conjunction: Definition

The logical value of a **conjunction** depends on the logical values of its factors in a way which is express in the form of the following table (truth table).

Conjunction Table:

A	B	$(A \cap B)$
T	T	T
T	F	F
F	T	F
F	F	F

Disjunction

The word or is used in natural language in two different senses.

First: A or B is true if at least one of the statements A, B is true

Second: A or B is true if one of the statements A and B is true and the other is false

In mathematics and hence in logic, the word or is used in the **first sense**

Disjunction: Definition

We adopt the convention that a **disjunction** $(A \cup B)$ is true if at least one of the formulas A , B is true

Disjunction Table:

A	B	$(A \cup B)$
T	T	T
T	F	T
F	T	T
F	F	F

Negation: Definition

The **negation** of a true formula is a false formula, and the negation of a false formula is a true formula

Negation Table:

A	$\neg A$
T	F
F	T

Implication: Motivation and Definition

The semantics of the statements in the form

if A, then B

needs a little bit more discussion.

In everyday language a statement *if A, then B* is interpreted to mean that B can be **inferred** from A.

In mathematics its interpretation differs from that in natural language

Implication: Motivation and Definition

Consider the following

Theorem

For every natural number n,

if 6 DIVIDES n, then 3 DIVIDES n

The theorem is **true** for any natural number, hence in particular, it is **true** for numbers 2, 3, 6

Consider number 2

The following proposition is **true**

if 6 DIVIDES 2, then 3 DIVIDES 2

It means an implication $(A \Rightarrow B)$ in which A and B are **false** is interpreted as a **true** statement

Implication: Motivation and Definition

Consider now a number 3

The following proposition **is true**

$$\text{if } 6 \text{ DIVIDES } 3, \quad \text{then } 3 \text{ DIVIDES } 3,$$

It means that an implication $(A \Rightarrow B)$ in which A is **false** and B is **true** is interpreted as a **true statement**

Consider now a number 6

The following proposition is **true**

$$\text{if } 6 \text{ DIVIDES } 6, \quad \text{then } 3 \text{ DIVIDES } 6.$$

It means that an implication $(A \Rightarrow B)$ in which A and B are **true** is interpreted as a **true statement**

Implication: Motivation and Definition

One more case.

What happens when in the implication $(A \Rightarrow B)$ the formula A is **true** and the formula B is **false**

Consider a sentence

if 6 DIVIDES 12, then 6 DIVIDES 5.

Obviously, this is a **false statement**

Implication: Definition

The above examples justify adopting the following definition of a semantics for the implication $(A \Rightarrow B)$

Implication Table:

A	B	$(A \Rightarrow B)$
T	T	T
T	F	F
F	T	T
F	F	T

Equivalence Definition

An equivalence $(A \Leftrightarrow B)$ is **true** if both formulas A and B have the same logical value

Equivalence Table:

A	B	$(A \Leftrightarrow B)$
T	T	T
T	F	F
F	T	F
F	F	T

Extensional Connectives

Extensional connectives are the connectives that have the following property:
the logical value of the formulas form by means of these connectives and certain given formulas depends only on the logical value(s) of the given formulas

All classical **propositional connectives**

$$\neg, \ \cup, \ \cap, \ \Rightarrow, \ \Leftrightarrow$$

are **extensional**

Propositional Connectives

Remark

In everyday language there are expressions such as

"I believe that", "it is possible that", " certainly", etc....

They are represented by some **propositional connectives** which **are not extensional**

They do not play any role in **mathematics** and so are not discussed in **classical logic**, they belong to **non-classical** logics

Connectives Symbols

Other Notations

Negation	Disjunction	Conjunction	Implication	Equivalence
$-A$	$A \cup B$	$A \cap B$	$A \Rightarrow B$	$A \Leftrightarrow B$
NA	DAB	CAB	IAB	EAB
$\overline{A}$	$A \vee B$	A & B	$A \rightarrow B$	$A \leftrightarrow B$
$\sim A$	$A \vee B$	$A \cdot B$	$A \supset B$	$A \equiv B$
A'	$A + B$	$A \cdot B$	$A \rightarrow B$	$A \equiv B$

The first notation is the closest to ours and is drawn mainly from the algebra of sets and lattice theory.

The second comes from the Polish logician **J. Łukasiewicz** and is called the Polish notation

The third was used by **D. Hilbert.**

The fourth comes from **Peano** and **Russell**

The fifth goes back to **Schröder** and **Pierce**

All Extensional Connectives

There are many other extensional propositional connectives!

Here is a table of **all unary** connectives

A	$\triangledown_1 A$	$\triangledown_2 A$	$\neg A$	$\triangledown_4 A$
T	F	T	F	T
F	F	F	T	T

All Extensional Connectives

Table of all binary connectives:

A	B	$(A \circ_1 B)$	$(A \cap B)$	$(A \circ_3 B)$	$(A \circ_4 B)$
T	T	F	T	F	F
T	F	F	F	T	F
F	T	F	F	F	T
F	F	F	F	F	F
A	B	$(A \downarrow B)$	$(A \circ_6 B)$	$(A \circ_7 B)$	$(A \leftrightarrow B)$
T	T	F	T	T	T
T	F	F	T	F	F
F	T	F	F	T	F
F	F	T	F	F	T
A	B	$(A \circ_9 B)$	$(A \circ_{10} B)$	$(A \circ_{11} B)$	$(A \cup B)$
T	T	F	F	F	T
T	F	T	T	F	T
F	T	T	F	T	T
F	F	F	T	T	F
A	B	$(A \circ_{13} B)$	$(A \Rightarrow B)$	$(A \uparrow B)$	$(A \circ_{16} B)$
T	T	T	T	F	T
T	F	T	F	T	T
F	T	F	T	T	T
F	F	T	T	T	T

Functional Dependency Definition

Definition

Functional dependency of connectives is the ability of defining some connectives in terms of some others

classical propositional connectives can be defined in terms of disjunction and negation.

Two binary connectives: $\downarrow$ and $\uparrow$ suffice, each of them separately, to define **all classical connectives**, whether unary or binary

Functional Dependency

The connective $\uparrow$ was discovered in 1913 by **H.M. Sheffer**, who called it alternative negation

Now it is often called a Sheffer's connective

The formula

$A \uparrow B$ reads: not both A and B.

Negation $\neg A$ is defined as $A \uparrow A$.

Disjunction $(A \cup B)$ is defined as $(A \uparrow A) \uparrow (B \uparrow B)$

Functional Dependency

The connective $\downarrow$ was termed by **J. Łukasiewicz** a joint negation

The formula

$A \downarrow B$ reads: neither A nor B.

It was proved in 1925 by **E. Żyliński** that no propositional connective other than $\uparrow$ and $\downarrow$ suffices to define all the remaining classical connectives

Write the proof as an exercise