CISC 204 Class 1

Course Overview and Introduction to Natural Deduction

Text Correspondence: pp. 1–6

Main Concepts:

- *Proposition: logical statement that is true or false*
- *Atomic proposition: single symbol*
- *Operator: logical connective*
- *WFF: well formed formula, a proposition that satisfies rules of construction*
- *Formula: a WFF represented as a Greek character such as ϕ or ψ*

This course is an introduction to formal logic, which is the study of abstract deductive reasoning. We will use symbols to represent elementary "things" that are either true or false, and study ways of proving that complex statements are true or false within a given logical system.

1.1 Logic From an Historical Perspective

- *c. 800 BC:* Greeks rediscovered writing.
- *c. 600 BC:* Mathematics as a deductive science was introduced by Thales of Miletus.
- *c. 500 BC:* Pythagoras introduced natural numbers.
- *c. 350 BC:* Aristotle introduced logic as a way of determining whether reasoning was correct. Introduced the *syllogism*:

  All birds can fly
  Tweety is a bird
  Therefore Tweety can fly

An example of a logical argument:

> If the train arrives late and there are no taxis at the station, then John is late for his meeting. John is not late for his meeting. The train did arrive late. *Therefore*, there were taxis at the station.

One symbolic form of this logical argument:

p: The train arrives late
q: There are taxis at the station
r: John is late for his meeting

If p and not q then r; not r; p
Therefore, q

An argument that is not logical:

If a course is fun to learn then you will do well on the exam. Jane did well on the CISC 204 exam, therefore it was fun to learn.

1.2 Propositional Logic

In this course, a *proposition* is a declarative sentence that can take on a value of true or false. Examples of propositions include:

- The sum of 2 and 2 is equal to 5
- Today is Thursday
- Every rational number greater that 2 can be expressed as the product of prime numbers
- The moon is made of green cheese

1.2.1 Syntax for Propositional Logic

- Lower-case Roman letters $p, q, r, \ldots$ are used to denote *atomic propositions*. For example,

  p: It is raining
  q: It is winter
  r: It is cold

- We can form *complex sentences*, or *propositions*, using logical connectives:

  $\neg p$ denotes not p, or it is not the case that p is true
  $p \vee q$ denotes that p is true or q is true
  $p \wedge q$ denotes that p is true and q is true
  $p \rightarrow q$ denotes that if p is true then q is true

- We have two special symbols: $\top$ for "always true" and $\bot$ "always false"

The specific rules for constructing a proposition are provided in the text book for this course.

1.2.2 Bracketing Conventions:

$\neg$ binds more tightly than $\wedge$ or $\vee$, and the latter two more tightly than $\rightarrow$. Implication $\rightarrow$ is right-associative:

$p \rightarrow q \rightarrow r$ denotes $p \rightarrow (q \rightarrow r)$

When in doubt, it is best to use brackets to clarify your intention.

1.2.3 Logical Formula:

In the textbook, §1.3 provides a strict definition of "logical formula", specifically as a well-formed formula or WFF. For now, we can understand that a formula is an atomic proposition, or is composed from one or more logical formulas using logical connectives.

We will use a small number of lower-case Greek letters to represent logical formulas. We will most commonly use ϕ (phi) and ψ (psi), and occasionally will use χ (chi) or will subscript a formula letter when that will make our intentions clearer.

For example, if ϕ is a logical formula and ψ is a logical formula, then $\phi \vee \psi$ is a logical formula. This allows us to recursively define the concept of a logical formula.

Definition: sequent

Assume that we have a set of formulas $\phi_1, \phi_2, \ldots_n$, which we will call the *premises*. Also assume we have a formula ψ, which we will call the *conclusion*.

We say that a *sequent* is a set of assumptions that are postulated to lead to a conclusion. For the assumed formulas, we would write this as

$$\phi_1, \phi_2, ...\phi_n \vdash \psi$$

A key concept in this course is whether a sequent is syntactically *valid*, which means that if we can find a *proof* for ψ given the premises $\phi_1, \phi_2, \ldots \phi_n$. For example,

$$p \vee q, \neg q \vdash p$$

is syntactically valid. (Later in the course, we will consider the "meaning", or *semantics*, of a sequent.)

1.3 Natural Deduction

In logic, there are many ways that a logical system can be constructed. One common way is to use *axioms*, which are logical formulas that are taken to be always true, and a *proof rule*. This way is often used in mathematical logic and related studies. Ideally, from the axioms, only "true" statements can be deduced or proved.

Another alternative is to use few (or zero!) axioms and instead to use numerous proof rules. The textbook for this course uses zero axioms and quite a few rules of deduction. This is called *natural deduction* because it mimics how many people naturally think. It has a further advantage when applied to computer code: natural deduction makes it easy to prove assertions about how a piece of code will work.

Some of the elementary proof rules in natural deduction might seem "obvious" but, to allow us to perform deductions, we need to clearly state the rules.