Discrete Mathematics: Chapter 1

Steven Schmatz, University of Michigan: College of Engineering

September 1, 2014

stevenschmatz@gmail.com

Chapter 1: The Foundations: Logic and Proofs

- A *proof* is a correct mathematical argument.
- A *theorem* is a mathematical statement that is proven to be true.

1.1: Propositional Logic

Propositions

Logic gives us rules to distinguish between valida nd invalid mathematical arguments. A *proposition* is a declarative sentence that is either true or false, but not both. *Propositional variables* are variables that represent propositions, such as $p, q, r, s \ldots$, just as letters are used to denote numerical variables.

- $\neg p$ represents the *negation of* p.
- $p \land q$ represents the *logical conjunction of* p *and* q, essentially an `and` statement in traditional programming languages.
- $p \lor q$ represents the *logical disjunction of* p *and* q, essentially an `or` statement in traditional programming languages.

The *or* statement can be in two forms: the *inclusive or*, or the *exclusive or*. The inclusive or is a disjunction when **at least one** of the propositions is true. The exclusive or, $p \oplus q$, is a disjunction which is true only when one of the two propositions is true.

Conditional statements

Let p and q be propositions. p is known as the *hypothesis* (or *antecedent* or *premise*), while q is known as the *conclusion* (or *consequence*).

The *conditional statement* (also known as an implication) $p \to q$ is the proposition: if p, then q. This statement evaluates to true in all cases except when p is true and q is false.

If you think of p to be a strict subset of q, then whatever is evaluated true in p should also be true in q. Hence, if p is evaluated true, so should p be evaluated true. If q is *not* true, then this statement is impossible and $p \to q$ is evaluated false.

Converse, contrapositive, and inverse

- The *converse* of a proposition $p \to q$ is $q \to p$. The converse never has the same truth value as the proposition for all possible truth values of p and q.

- The *contrapositive* of a proposition $p \to q$ is $\neg q \to \neg p$. A contrapositive statement **always** has the same truth statement as the proposition.
- The *inverse* of a proposition $p \to q$ is $\neg p \to \neg q$. The inverse never has the same truth value as the proposition for all possible truth values of p and q.

Two compound statements are called *equivalent* when they have all the same truth values for all values of p and q.

Propositional logic is different from if-then statements both in the English language and in computer programming:

- In English, the statement 'If Juan gets a smartphone, then $2 + 3 = 6$' is very obviously false. However, in propositional logic, this conditional statement would be completely valid.
- In computer programming, if-then statements take the form of `if` p `then` S, where p is a propositional statement and S is a segment of code.

Mathematical propositions are much more general than in either of these two cases.

Biconditionals

Biconditional statements, $p \leftrightarrow q$, also known as "p if and only if q", are only true when both p and q are of the same truth value. This can also be expressed as the following: the statement is only true when $p \to q$ and $q \to p$.

Compound statements

With these four logical connectives – conjunctions, disjunctions, conditional statements, and biconditional statements – one can create complex logical statements, such as the one below:

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

To correctly interpret these statements, one needs the order of precedence.

1. The negation operator $\neg$.
2. The conjunction operator $\wedge$.
3. The disjunction operator $\vee$.
4. The implication operator $\to$.
5. The biconditional operator $\leftrightarrow$.

Logic and binary operations

A *bit*, or “binary digit”, is a value which may either be 0 or 1. A variable that stores a bit is known as a *boolean variable*. All of the logical operators can be applied to Boolean variables.