COMP 409, Lecture 1

January 16, 2003

1 Logic in Computer Science

The following are uses of Logic in Computer Science:

1. * Computational Complexity
2. * Databases
3. * Design Verification
4. AI reasoning
5. Circuit Design
6. Programming Languages / Semantics
7. Security - proof carrying code

2 Basics of Propositional Logic

Truth Follows Form. Consider the following example:

- All men are mortal
- + <u>All greeks are men</u>
- All greeks are mortal

By replacing the key words this statement can be translated to:

- All bojums are slithy
- + <u>All toves are bojums</u>
- All toves are slithy

The point is *Mathematics is abstract*! We can abstract word problems. We can use symbols.

Definition 1 Two-Valued Logic - *things are true or false and there is nothing inbetween.*

Definition 2 Digital Abstraction - *Electrical Engineers' job to give us digitial appearance from analog signals: 0 and 1.*

Definition 3 Propositional Logic *or* **Sentential Logic** *or* **Boolean Logic** - *Logic based on statements which must be true or false. or Logic of Propositions.*

Example:

- If Socrates is human, then Socrates is mortal
- If Socrates is Greek, then Socrates is human
- Socrates is Greek
- Therefore, Socrates is mortal

In the above example the **Propositions** are:

- "Socrates is human" = P
- "Socrates is mortal" = Q
- "Socrates is greek" = R

Now by substituting symbols, the set of statements looks like:

- If P, then Q
- If R, then P
- R
- Therefore, Q

3 Propositional Connections

Once we have these symbols we can develope an algebra to connect them.
Binary Connections

- iff $\leftrightarrow$
- if then $\rightarrow$
- and $\wedge$
- or $\vee$
- and others {xor, nand}

Unary Connections

- not $\neg$

<u>Ternary Connections</u>

- iff then else

We will just use $\{\leftrightarrow, \rightarrow, \wedge, \vee, \neg\}$ for now.
We also need a language or alphabet to work over:

- Propositions – $p, q, r, p_0, p_1, p_2, \ldots = Prop$
- *Connectives* – $\wedge, \vee, \rightarrow, \leftrightarrow, \neg$
- *Parenthases* – (,)

4 Expressions

Definition 4 *An* **expression** *is a string of alphabet symbols. Not all expressions are of interest, e.g.,* $\leftrightarrow)p($

Definition 5 *A* **formula** *is an expression that follows a particular form. The set of formulas is the smallest set,* <u>*Form*</u>*, such that:*

1. $Prop \subseteq Form$
2. Closure*:*
   (a) *If* $\varphi \in Form$*, then* $(\neg\varphi) \in Form$
   (b) *If* $\varphi, \psi \in Form$*, then* $(\varphi \circ \psi) \in Form$*, where* $\circ \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$

Now we need to prove that a smallest formula exists.

Lemma 1 *There exists a smallest set Form that contains Prop and satisfies the closure property.*

<u>Proof</u> let $Form\prime = \bigcap\{S : Prop \subseteq S$ and S satisfies closure$\}$
Claim 1 $Form\prime$ is the intersection of sets that contain $Prop$, so $Prop \subseteq Form\prime$. Suppose φ, $\psi \in Form\prime$, then φ, $\psi \in S$, for all S containing $Prop$ and satisfies closure. So $(\varphi \circ \varphi) \in S$. So $(\varphi \circ \psi) \in Form\prime$.
Now show this is the smallest set. Suppose S contains $Prop$ and satisfies closure. By definition, $Form\prime \subseteq S$. We can conclude $Form\prime = Form$. QED

Definition 6 $Form_0 = Prop$*.* $Form_{i+1} = Form \cup \{(\neg\varphi)\colon \varphi \in Form\} \cup \{(\varphi \circ \psi)\colon \varphi, \psi \in Form\}$*.*

Lemma 2 $Form = \bigcup_{i=0}^{\infty} Form$

Proof Let $Form\prime$ be $\bigcup_{i=0}^{\infty} Form$.

Show $Form \subseteq Form\prime$: Suffice to show that $Form\prime$ contains Prop and satisfies closure. $Prop = Form_0 \subseteq Form\prime$.

Let φ, $\psi \in Form\prime$: Exists i s.t. $\varphi \in Form_i$ and exists j s.t. $\psi \in Form_j$.

Let $k = max\{i, j\}$ then φ, $\psi \in Form_k$.

So $(\psi \circ \varphi) \in Form_{k+1} \subseteq Form\prime$.

Clearly, $Form'$ contains $Prop$ and satisfies closure. Since $Form$ is miniamal such set, we have that $Form = Form'$. QED