Discrete Mathematical Structures
CS 3233 Lecture Four

Prof. William Winsborough
January 24, 2008

Business

- Questions???
- Turn in Homework 2
- Homework 3 due Thursday 1/31:
  - Section 1.4: 2a, 2c, 4c-f, 10 (except h), 24a, 24d, 46
    - Work 4a, 4b, 10h together
    - In 46, ***explain your answers*** and assume the usual interpretation for arithmetic and relational operators
  - Section 1.5: 4, 8, 16
  - Section 1.6: 2 (natural-language proof), 12, 18
  - Section 1.7: 4, 6, 8
- Read for Tuesday 9/18
  - 1.4, 1.5 and 1.6
  - 1.7 through example 14 (p.95)

Predicate Formulas as Specifications

- For the domain of integers, is ∀x.(x>3) true or false?
  - How about ∃x.(x>3) ?
  - Note: ">" is the predicate symbol here
  - X>3 is another way of writing >(x,3)
- What does ∀x ∀y.(x+y = y+x) mean?
  - Is it true?
  - What is the predicate here?
  - In logic, "+" is called a *function symbol* (term not introduced in the text)

Formal Notion of "Interpretation"

- An *Interpretation* of a formula consists of:
  - A domain of discourse
    - *E.g.,* bit values (Boolean), natural numbers, integers, rationals, reals, vectors of same
  - A meaning for each predicate symbol
    - *I.e.,* a propositional function
    - *E.g.,* "<" refers to "less than" over bit vectors
  - A meaning for each constant symbol
    - *E.g.,* "1" refers to the natural number 1, "**T**" denotes the Boolean value *true*, "∅" denotes the empty set, "λ" denotes the empty string
  - A meaning for each function symbol
    - *E.g.,* "+" refers to addition of natural numbers, "•" denotes dot product over vectors of real numbers

Predicate Formulas in Software Specifications

- One of the primary uses of predicate-logic formulas in computer science is specification of software
  - Software behavior can thought of in terms of an intended interpretation
    - E.g., a library for high-precision arithmetic
  - Formulas serve to specify requirements of software behavior

Exercise: In Which Numeric Domains Does each of the Following Hold?

- Domains
  - Z – The integers
  - N – The natural numbers (non-negative integers)
  - R^+ – The positive reals
  - $R^+ \cup \{0\}$ – The non-negative reals
- Formulas ("<" and "≤" have usual interpretation)
  - ∀x.∃y.y<x
  - ∃x.∀y.x ≤ y
  - ∀x.∀z.(x<z →∃y.(x<y)∧(y<z))

Theorems and Proofs

- A *theorem* is a statement (such as a formula) that can be shown to be true in all cases (a tautology)
- A proof is a demonstration that a statement is a theorem
- Example methods of proof
  - Construction of truth tables
  - Use of equivalences
    - By using these alone, can prove only logical equivalences
  - More general rules of inference

Important Related Terminology

- *Result*: often used to mean a theorem
- *Proposition*: a simple theorem, often presented without proof
- *Lemma*: a theorem whose main utility lies in helping to prove other, more interesting theorems
- *Corollary*: a theorem that follows easily from another more general theorem
- *Conjecture*: a statement that you suspect is true but that you do not yet have a proof for

Rules of Inference for Propositional Logic

- A general, systematic method of proving formulas
- See Table 1 p.66
  - Known equivalences can also be used in proofs
- Use rules of inference to show
  - These hypotheses:
    - If it does not rain or if it is not foggy, the sailing race will be held and the lifesaving demonstration will take place
    - If the race is held, the trophy will be awarded
    - The trophy was not awarded
  - Imply this conclusion:
    - It rained

Rules of Inference for Universally Quantified Statements

- Universal instantiation

$$\frac{\forall x.\ p(x)}{p(c)}$$

  - For any c
- Universal generalization

$$\frac{p(c)}{\forall x.\ p(x)}$$

  - Must show p(c) for arbitrary c

Rules of Inference for Existentially Quantified Statements

- Existential instantiation

$$\frac{\exists x.\ p(x)}{p(c)}$$

  - c must be a new name (constant) that does not appear earlier in the proof
- Existential generalization

$$\frac{p(c)}{\exists x.\ p(x)}$$

  - c can be any name

Universal Modus Ponens

- Combines propositional modus ponens with universal instantiation:

$$\frac{\forall x.\Phi(x)\rightarrow\Psi(x) \qquad \Phi(c)}{\Psi(c)}$$

- Practice problems 14 and 15, p.73