CSE 150: Foundations of Computer Science
Course Summary

R. Sekar

Propositions

- Propositions and theorems
- Propositional variables and operators
- Validity, satisfiability and equivalence
- Truth tables: construction and use
  - Showing equivalence of propositions using the truth table
- Equivalences
  - Expressing one logical operator in terms of others
  - De Morgan's Laws
  - Establishing equivalence of propositions
- Normal forms: DNF and CNF
  - Converting to normal form
  - CNF and the SAT problem

Proofs

- Axioms and inference rules
  - Formal proofs
    - machine-checkable proofs and automated theorem proving
- Proof techniques
  - Proving an implication
    - Assume antecedent (hypothesis), show that the consequent holds
    - False hypothesis and Vacuous truth
    - Prove by establishing the contrapositive
    - Prove equivalence by proving both sides of implication
  - Proof by cases
  - Proof by contradiction
  - Proof by induction

Predicates

- Predicate Vs Proposition
- Variables, valuations, and evaluating predicates
- Satisfiability and Validity
- Quantifiers
  - Free and Bound variables
- Order of quantifiers
- Negating quantifiers
- Conversion between English and Logical Formulas

Sets

- Definition of sets
- Set builder notation
- Set operators: membership, subset, union, intersection, difference
  - Properties of set operators: commutativity, associativity, distributivity
- Set equality
- Universal set, set complement and De Morgan's Laws
- Power Set
- Cartesian Product and Tuples

Functions and Relations

- Binary Relations
  - Domain, codomain, support and range
  - Image and Graph
- Relations over multiple sets
- Properties of relations
  - function (≤ 1 arrow **out**)
  - total functions (≥ 1 arrow **out**)
  - one-to-one (injective, ≤ 1 arrow **in**)
  - onto (surjective, ≥ 1 arrow **in**)
  - bijective
- Composition ($f \circ g$, $R \circ S$, R^n)
- Inverse (R^{-1})

Relations and Graphs

- Directed graphs and binary relations from a set to itself
  - vertex (aka node) and edges
  - degree, equality of the sum of in- and out-degrees
- walks, paths, and relational composition
- closed walks and cycles
- shortest paths and distances in a graph
- directed acyclic graphs (DAGs) and topological sorting
- Euler and Hamiltonian tours
  - Euler's theorem
- Types of relations
  - Reflexive, Irreflexive, Symmetric, Antisymmetric, Transitive
  - Partial orders, Linear orders, Equivalences and Equivalence Classes
  - Closures (Reflexive, Symmetric, Transitive)

Set Cardinality

- Comparing cardinalities using maps
  - $|A| \geq |B|$ iff A *surj* B
  - $|A| \leq |B|$ iff A *inj* B
  - $|A| = |B|$ iff A *bij* B
- Definition of infinite sets
- Countability
- Proving countability
  - Specify enumeration order and prove every element eventually appears
  - Using injection/surjection/bijection
  - Using closure under union/product/intersection/subset
- Uncountable sets
- Proving uncountability
  - Diagonalization
  - Power sets and Cantor's theorem

Summation

- Perturbation Method
  - Example 1: Geometric Progression: $\sum_{i=0}^{n} x^i = \frac{x^{n+1}-1}{x-1}$
  - Example 2: Arithmetic Progression: $\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$
  - Example 3: Sum of i^k: $\sum_{i=1}^{2} = \frac{n(n+1)(2n+1)}{6}$ $\qquad\qquad$ $\sum_{i=1}^{3} = \left(\frac{n(n+1)}{2}\right)^2$
- Using differentiation
  - Example 4: Arithmetico Geometric Progression: $\sum_{i=1}^{n-1} ix^{i-1} = \frac{(n-1)x^n - nx^{n-1}+1}{(1-x)^2}$
- Using integration
  - Example 5: $1 + \frac{2}{3}(n^{3/2} - 1) \leq \boxed{\sum_{x=1}^{n} \sqrt{x}} \leq \sqrt{n} + \frac{2}{3}(n^{3/2} - 1)$
  - Example 6: Factorial: $n\ln(n) - n + 1 \leq \boxed{\sum_{i=0}^{n} \ln(i)} \leq n\ln(n) - n + 1 + \ln(n)$
  - Example 7: Hanging blocks: $\frac{1}{n} + \ln(n) \leq \boxed{\sum_{i=1}^{n} \frac{1}{i}} \leq \ln(n) + 1$

Counting

- Bijections and counting
- Counting rules
  - Union: Sum and Inclusion-Exclusion
  - Cartesian products: Product and Generalized product rule
  - Division rule
- Permutations and combinations
- Counting using binary sequences: interpret each sequence as a procedure for picking a unique element
  - # of subsets, # of subset sequences
  - integer partitioning, increasing sequences, balls & bins w/ no restrictions on balls/bin
  - Donut problem: number of distinct sets using elements of k different types
  - choosing books
- Pigeonhole principle

Probability

- Random variables, outcomes, sample space, events
- Uniform probability spaces, counting and probability
  - Birthday problem
- Probability rules from set theory
  - Sum, complement, difference, inclusion-exclusion
  - monotonicity, union bound
- Conditional probability
  - Formulas
  - tree diagrams and the 4-step method
  - medical testing
- Independence

Recurrences and Recursion

- Computing using recursive functions
  - Examples: factorial, fibonacci, Tower of Hanoi
- Asymptotic complexity and O-notation
- Solving recurrences
  - Plug-and-chug
  - Linear recurrences
  - Master theorem

Recursive functions, data types and programming

- Programming with functions from mathematics
- Induction and recursion
  - proving correctness of recursive definitions using induction
  - inductive data type definitions
- Standard ML
  - Values
  - Functions
  - Data types: integers, reals and strings; tuples and lists
  - Pattern-matching
- Programming with lists

CSE 150 Concepts in SML

- Propositions
- Predicates and quantifiers
- Sets: union, intersection, product, differences
- Relations
  - Graphs
  - Binary trees and search trees
- Sequences
- Recurrences and summations
- Counting
- Random numbers and probability

Relationship to Future CS Courses

- Propositional and predicate logic: Computer organization (220), CSE 304 (Compilers), CSE 305 (databases), Theory of computation (350), Algorithms (385), Software Engineering (CSE 416), ...
- Proofs: everywhere, especially CSE 350 and 385
- Sets and Relations: CSE 305, 350, 385, ...
- Graphs: CSE 385, 310 (networks), AMS 301
- Summations, Recurrences, Cardinality: CSE 350, 385, AMS 301
- Counting and probability: CSE 385, AI/ML courses, AMS 301 and 310
- Recursion: Every course, starting with 160 and 260