1 Logic

- Sentences
  - not $\neg$
  - and $\wedge$
  - or $\vee$
  - tautology $a = a$
  - contradiction $a \neq a$
- Truth tables
- Laws of equivalence
  - Commutative $a \wedge b \equiv b \wedge a$
  - Associative $(a \wedge b) \wedge c \equiv a \wedge (b \wedge c)$
  - Distributive $a \wedge (b \vee c) \equiv (a \wedge b) \vee (a \wedge c)$
  - De Morgan's $\neg(a \wedge b) \equiv \neg a \vee \neg b$
- Proof of De Morgan's with truth table
- Conditional statements $a \to b \equiv \neg a \vee b$
  - Hypothesis
  - Conclusion
  - Contrapositive $a \to b \equiv \neg b \to \neg a$
  - Converse $b \to a$
  - Inverse $\neg a \to \neg b$
  - Biconditional statements $a \leftrightarrow b$
- Arguments
  - Modus ponens (method of affirming) $a \to b \quad a \quad \therefore b$
  - Modus tollens (method of denying) $a \to b \quad \neg b \quad \therefore \neg a$

2 Predicates and Quantified statements

- Predicate
  - Domain $D = \{1, 2, 3, 4, 5, 6, 7, 8, 9\} = \{1, \ldots, 9\}$
  - Set-builder $\{x \in D \mid P(x)\}$
  - Truth set $\{1, \ldots, 4\}$
- Quantifiers
  - Universal quantifier $\forall$
  - Existential quantifier $\exists$
- Universal Conditional Statement $\forall x, P(x) \rightarrow Q(x) \equiv P(x) \Rightarrow Q(x)$
- Arguments
  - Modus ponens $\forall x, P(x) \rightarrow Q(x) \quad P(a) \quad \therefore Q(a)$
  - Modus tollens $\forall x, P(x) \rightarrow Q(x) \quad \neg Q(a) \quad \therefore \neg P(a)$
- Multi-quantified statements $\forall x \in D, \exists y \in E, P(x, y)$
- Laws of multi-quantified statements
  - De Morgan's $\neg(\forall x \in D, \exists y \in E, P(x, y)) \equiv \exists x \in D, \forall y \in E, \neg P(x, y)$

3 Sequences, Induction and Recursion

- Sequences
  - Definition $D : f(x)$
  - Properties
- Summations $\sum$
- Telescopic sums $\sum_{k=1}^{n} \frac{1}{k(k+1)} = \frac{1}{1} - \frac{1}{n+1}$
- Product $\prod$
- Theorems
  - Adding and removing a final term
  - $\sum_{k=m}^{n} a_k + \sum_{k=m}^{n} b_k = \sum_{k=m}^{n} a_k + b_k$
  - $c\dot{\sum}_{k=m}^{n} a_k = \sum_{k=m}^{n} c\dot{a}_k$
  - $\prod_{k=m}^{n} a_k + \prod_{k=m}^{n} b_k = \prod_{k=m}^{n} a_k + b_k$
- Factorials $n!$
- Combinations $\binom{n}{r} = \frac{n!}{r!(n-r)!}$
- Induction
  - $a, k \in \mathbb{Z}$
  - If $P(a)$
  - and $\forall k \geq a, P(k) \rightarrow P(k+1)$
  - then $\forall n \geq a, P(n)$
- Prove sum of first n integers by induction
- Recursion

4 Regular Expressions and Finite-state Automata

- Chomsky
  - Regular languages $\begin{matrix} A \to a \\ A \to aB \end{matrix}$
  - Context-free languages $A \to \alpha$
  - Context-sensitive languages $\alpha A \beta \to \alpha \gamma \beta$
  - Turing-complete languages $\alpha A \beta \to \gamma$
- Formal languages
  - Alphabet Σ
  - String over Σ
  - Language L over Σ
- Combining languages
  - Concatenation $LL' = \{xy \mid x \in L \land y \in L'\}$
  - Union $L \cup L' = \{x \mid x \in L \lor x \in L'\}$
  - Kleene closure $L^* = \{x \mid \text{is a concatenation of strings in L}\}$
- Regular Expressions
  - Base/terminals $\emptyset, \epsilon, x \mid x \in \Sigma$
  - Recursion/non-terminals
    - (rs) - r concatenated with s
    - $(r \mid s)$ - r or s
    - r^* - $r^* \in \{\epsilon, r, rr, rrr, \ldots\}$
- Finite-state automaton
  - Input alphabet Σ
  - States S
  - Initial state $s_0, s_0 \in S$
  - Final states F
  - Next-state function $N : S \times \Sigma \to S$
- Eventual-state function $N^* : S \times \Sigma^* \to S$
- Draw $a(b|cd)^*e$
- Regular languages
  - can be defined by a regular expression

    $\Updownarrow$
  - can be accepted by a finite-state automata

5 Set Theory

- Notation
  - Set-roster notation $A = \{1, 2, 3\} \qquad B = \{10, 11, 12, \dots, 119\}$
  - Set-builder notation $M = \{x \in S \mid P(x)\}$
  - The empty set $\emptyset = \{\}$
  - Powerset $\mathscr{P}(\{a, b\}) = \{\emptyset, \{a\}, \{b\}, \{a, b\}\}$
- Operations
  - Membership $e \in M$
  - Intersection $A \cap B$
  - Union $A \cup B$
  - Difference $A - B$
  - Complement A^C
- Subset $A \subseteq B \Leftrightarrow \forall x \in A, x \in B$
  - Not a subset $A \nsubseteq B \Leftrightarrow \exists x \in A, x \notin B$
  - Proper subset $A \subset B \Leftrightarrow \forall x \in A, x \in B \wedge \exists x \in B, x \notin A$
  - Equality $A = B \Leftrightarrow A \subseteq B \wedge B \subseteq A$
  - Transitivity $A \subseteq B \wedge B \subseteq C \rightarrow A \subseteq C$
- Disjoint sets $A \cap B = \emptyset$
  - Partitions $\{A_1, A_2, \dots, A_n\}$ of set A
  - $A = \bigcup_{i=1}^{n} A_i$
  - $\forall a, b \in \{1, 2, \dots, n\}, A_a \cap A_b = \emptyset \vee a = b$
- Laws
  - Commutative $A \cap B = B \cap A$
  - Associative $(A \cap B) \cap C = A \cap (B \cap C)$
  - Distributive $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$
  - De Morgan's $(A \cup B)^C = A^C \cap B^C$
- Ordered pair $(a, b) = \{\{a\}, \{a, b\}\}$
- Cartesian product $\begin{array}{c} A \times B = \{(a, b) \mid a \in A \wedge b \in B\} \\ A \times B \times C = \{(a, b, c) \mid a \in A \wedge b \in B \wedge c \in C\} \end{array}$
- Relations $R \subseteq \{(x, y) \in A \times B\}$
- Functions $\begin{array}{c} \forall x \in A, \exists y \in B, (x, y) \in F \\ \forall x \in A \wedge y \in B \wedge z \in B, ((x, y) \in F \wedge (x, z) \in F) \rightarrow y = z \end{array}$

6 Relations

- Definition
  - When R is a relation $x \, R \, y$
  - $R \subseteq \{(x, y) \in A \times B\}$
  - $R^{-1} = \{(x, y) \in B \times A \mid (x, y) \in R\}$
- Properties
  - Reflexivity $\forall x \in A, (x, x) \in R$
  - Symmetry $\forall x, y \in A, (x, y) \in R \rightarrow (y, x) \in R$
  - Transitivity $\forall x, y, z \in A, (x, y) \in R \wedge (y, z) \in R \rightarrow (x, z) \in R$
  - Equivalence relation
- Relation induced by partition $x \, R \, y \Leftrightarrow \exists A_i, x \in A_i \wedge y \in A_i$
  - Reflexive $(x \, R \, x)$
    $\exists A_i, x \in A_i \wedge x \in A_i$ by definition of $x \, R \, x$
  - Symmetric $(x \, R \, y \rightarrow y \, R \, x)$
    $\exists A_i, x \in A_i \wedge y \in A_i$ by definition of $x \, R \, y$
  - Transitive $(x \, R \, y \wedge y \, R \, z \rightarrow x \, R \, z)$
    $\exists A_i, x \in A_i \wedge y \in A_i$ by definition of $x \, R \, y$
    $\exists A_i, y \in A_i \wedge z \in A_i$ by definition of $y \, R \, z$
    $A_i \cap A_j = \emptyset \vee A_i = A_j$
    $y \in A_i \wedge y \in A_j \Leftrightarrow y \in A_i \cap A_j \Rightarrow A_i \cap A_j \neq \emptyset$
    $A_i = A_j \Rightarrow x \in A_i \wedge z \in A_i$ the definition of $x \, R \, z$
- Antisymmmetry
  - Anitsymmetric $\forall a, b \in A, a \, R \, b \wedge b \, R \, a \rightarrow a = b$
  - Not antisymmetric $\exists a, b \in A, a \, R \, b \wedge b \, R \, a \wedge a \neq b$
- Partial ordering
  - Reflexive, antisymmetric, transitive
  - $a \in A$ is maximum if: $\forall b \in A, b \preceq a \vee b \npreceq a$
  - $a \in A$ is greatest if: $\forall b \in A, b \preceq a$
  - $a \in A$ is minimum if: $\forall b \in A, a \preceq b \vee a \npreceq b$
  - $a \in A$ is least if: $\forall b \in A, a \preceq b$

7 Static Analysis

- Static analysis
  - Soundness: A static analysis is said to be sound if it rejects all faulty programs.
  - Completeness: A static analysis is said to be complete if all correct programs passes.
  - Process
    - Pre- and Post-conditions
    - "Run" code and update state
    - Compare state with postcondition
- Hoare logic
  - Hoare tripple $\{P\}C\{Q\}$
  - Skip commands $\{P\}skip\{P\}$
  - Command composition $\frac{\{P\}S\{Q\},\{Q\}T\{R\}}{\{P\}S;T\{R\}}$
  - Selection $\frac{\{B \wedge P\}S\{Q\},\{\neg B \wedge P\}T\{Q\}}{\{P\}if\ B\ then\ S\ else\ T\ endif\{Q\}}$
  - Iteration $\frac{\{B \wedge P\}S\{P\}}{\{P\}while\ B\ do\ S\ done\{\neg B \wedge P\}}$
- Design by contract
  - Idea
    - What does contract expect?
    - What does contract guarantee?
    - What does contract maintain?
  - Content
    - Input values and types
    - Return values and types
    - Error/Exception condition values and types
    - Side effects
    - Pre- and Post-conditions
    - Invariants