Foundations of Computer Science
Key Terms & Results

Peter Cappello
Department of Computer Science
University of California, Santa Barbara

The Foundations: Logic & Proofs

Terms

proposition: a declarative statement that is true or false, but not both

propositional variable: a variable that represents a proposition

$\neg p$ **(negation of** p**):** the proposition with truth value opposite to the truth value of p

logical operators: operators used to combine propositions

compound proposition: a proposition constructed by combining propositions using logical operators

$p \vee q$ **(disjunction of** p **and** q**):** the proposition "p or q," which is true if and only if at least 1 of p and q is true

$p \wedge q$ **(conjunction of** p **and** q**):** the proposition "p and q," which is true if and only if both p and q are true

$p \rightarrow q$ (p **implies** q**):** the proposition "if p then q," which is false if and only if p is true and q is false

$p \leftrightarrow q$ **(biconditional):** the proposition "p if and only if q," which is true if and only if p and q have the same truth value

$p \oplus q$ **(exclusive or of** p **and** q**):** the proposition "p XOR q," which is true when exactly 1 of p and q are true

converse of $p \rightarrow q$**:** $q \rightarrow p$

inverse of $p \rightarrow q$**:** $\neg p \rightarrow \neg q$

contrapositive of $p \rightarrow q$**:** $\neg q \rightarrow \neg p$

tautology: a compound proposition that always is true

contradiction: a compound proposition that always is false

predicate: the part of a sentence that attributes a property to the subject

propositional function: a statement containing 1 or more variables that becomes a proposition when each of its variables is assigned a value or is bound by a quantifier

domain (or universe) of discourse: the set of values a variable in a propositional function may take

$\exists x P(x)$ **(existential quantification of** $P(x)$**):** the proposition that is true if and only if there exists an x in the domain such that $P(x)$ is true

$\forall x P(x)$ **(universal quantification of** $P(x)$**):** the proposition that is true if and only $P(x)$ is true for every x in the domain

free variable: a variable not bound in a proposition function

bound variable: a variable that is quantified

scope of a quantifier: part of a statement where the quantifier binds its variable

argument: a sequence of statements

argument form: a sequence of compound propositions involving propositional variables

premise: a statement, in an argument or argument form, other than the final one

conclusion: the final statement in an argument or argument form

valid argument form: a sequence of propositions involving propositional variables where the truth of all the premises implies the truth of the conclusion

valid argument: an argument with a valid argument form

rule of inference: a valid argument form that can be used in the demonstration that arguments are valid

fallacy: an invalid argument form

theorem: a mathematical assertion that can be shown to be true

conjecture: a mathematical assertion proposed to be true, but that has not been proven

proof: a demonstration that a theorem is true

axiom: a basic assumption of a theory, assumed to be true, that can be used as a basis for proving theorems

lemma: a theorem used to prove other theorems

corollary: a proposition that can be proved as a consequence of a theorem

vacuous proof: a proof that $p \to q$ is true based on the fact that p is false

trivial proof: a proof that $p \to q$ is true based on the fact that q is true

direct proof: a proof that $p \to q$ is true that proceeds by showing that q must be true when p is true

proof by contraposition: a proof that $p \to q$ is true that proceeds by showing that p must be false when q is false

proof by contradiction: a proof that p is true based on the truth of $\neg p \to q$, where q is a contradiction

proof by cases: a proof decomposed into separate cases, where these cases cover all possibilities

without loss of generality: an assumption in a proof that makes it possible to prove a theorem by reducing the number of cases needed in the proof

counterexample: an element x such that $P(x)$ is false

constructive existence proof: a proof that an element with a specified property exists by explicitly finding such an element

nonconstructive existence proof: a proof that an element with a specified property exists that does not explicitly find such an element

RESULTS

- The following logical equivalences from Table 6:

  Double negation: $\neg(\neg p) \equiv p$

  Commutative:

  $p \vee q \equiv q \vee p$

  $p \wedge q \equiv q \wedge p$

  Associative:

  $(p \vee q) \vee r \equiv p \vee (q \vee r)$

  $(p \wedge q) \wedge r \equiv p \wedge (q \wedge r)$

  Distributive:

  $p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$

  $p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$

  DeMorgan's:

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

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

- The following equivalences of implication:

  $p \rightarrow q \equiv \neg p \vee q$

  $p \rightarrow q \equiv \neg q \rightarrow \neg p$

- The following equivalences of biconditional:

  $p \leftrightarrow q \equiv (p \rightarrow q) \wedge (q \rightarrow p)$

  $p \leftrightarrow q \equiv (p \rightarrow q) \wedge (\neg p \rightarrow \neg q)$

  $p \leftrightarrow q \equiv \neg(p \oplus q)$

- DeMorgan's laws for quantifiers

  $\neg\exists P(x) \equiv \forall x \neg P(x)$

  $\neg\forall P(x) \equiv \exists x \neg P(x)$

- The following rules of inference for propositional logic:

  Modus ponens: $[p \wedge (p \rightarrow q)] \rightarrow q$

  Hypothetical syllogism: $[(p \rightarrow q) \wedge (q \rightarrow r)] \rightarrow (p \rightarrow r)$

  Resolution: $[(p \vee q) \wedge (\neg p \vee r)] \rightarrow (q \vee r)$

- The rules of inference for quantified statements:

  Universal instantiation: If $\forall x P(x)$, then $P(c)$ for any c in the domain.

  Universal generalization: If $P(c)$ for arbitrary c, then $\forall x P(x)$

  Existential instantiation: If $\exists x P(x)$, then $P(c)$ for some c in the domain. *We however do not know which element in the domain c is.*

  Existential generalization: If $P(c)$ for some c, then $\exists x P(x)$

SET & FUNCTIONS

TERMS

set: a collection of distinct objects

paradox: a logical inconsistency

element, member of a set: an object in a set

$\emptyset$ **(empty set, null set):** the set with no members

universal set: the set containing all objects under consideration

Venn diagram: a graphical representation of a set or sets

$S = T$ **(set equality):** $\forall x(x \in S \leftrightarrow x \in T)$

$S \subseteq T$ **(**S **is a subset of** T**):** $\forall x(x \in S \rightarrow x \in T)$

$S \subset T$ **(**S **is a proper subset of** T**):** $S \subseteq T \wedge S \neq T$

finite set: a set with n elements, where n is a natural number

infinite set: a set that is not finite

$|S|$ **(the cardinality of** S**):** the number of elements in S

$P(S)$ **(the power set of** S**):** $\{s \mid s \subseteq S\}$

$A \cup B$ **(**A **union** B**):** $x \in A \cup B \leftrightarrow (x \in A \vee x \in B)$

$A \cap B$ **(**A **intersection** B**):** $x \in A \cap B \leftrightarrow (x \in A \wedge x \in B)$

$A - B$ **(**A **minus** B**):** $x \in A - B \leftrightarrow (x \in A \wedge x \notin B)$

$\overline{A}$ **(the complement of** A**):** $U - A$, where U is the universal set.

$A \oplus B$ **(symmetric difference of** A **and** B**):** $x \in A \oplus B \leftrightarrow (x \in A \oplus x \in B)$

membership table: a table displaying the membership of elements in sets

function from A **to** B**:** an assignment such that, $\forall a \in A$, a is assigned to exactly 1 element $b \in B$.

domain of f**:** the set A, where f is a function from A to B

codomain of f**:** the set B, where f is a function from A to B

b **is the image of** a **under f:** $b = f(a)$

a **is the pre-image of** b **under f:** $f(a) = b$

range of f: $\{b \mid \exists a \in A, f(a) = b\}$

onto function, surjection: f's range is its codomain: $\forall b \in B\ \exists a \in A, f(a) = b$

1-to-1 function, injection: $a \neq b \rightarrow f(a) \neq f(b)$

1-to-1 correspondence, bijection: a function that is a surjection and an injection.

inverse of f: when f is a bijection, its inverse, denoted f^{-1}, is the function $f^{-1}(b) = a$, where $f(a) = b$

$f \circ g$ **(composition of f and g):** the function that assigns $f(g(x)$ to x

$\lfloor x \rfloor$ **(floor function):** the largest integer not exceeding x

$\lceil x \rceil$ **(ceiling function):** the smallest integer greater than or equal to x

RESULTS

- The following set identities from Table 1:

  Complementation: $\overline{\overline{A}} = A$

  Commutative:
  $A \cup B = B \cup A$
  $A \cap B = B \cap A$

  Associative:
  $A \cup (B \cup C) = (A \cup B) \cup C$
  $A \cap (B \cap C) = (A \cap B) \cap C$

  Distributive:
  $A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$
  $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$

  DeMorgan's:
  $\overline{A \cup B} = \overline{A} \cap \overline{B}$
  $\overline{A \cap B} = \overline{A} \cup \overline{B}$

THE FUNDAMENTALS: ALGORITHMS & GROWTH OF FUNCTIONS

TERMS

algorithm: a finite sequence of precise instructions for performing a computation or solving a problem.

$f(x)$ **is** $O(g(x))$**:** the fact that $|f(x)| \leq C|g(x)|$, for all $x > k$, for some positive constants C and k.

$f(x)$ **is** $\Omega(g(x))$**:** the fact that $|f(x)| \geq C|g(x)|$, for all $x > k$, for some positive constants C and k.

$f(x)$ **is** $\Theta(g(x))$**:** the fact that $f(x)$ is $O(g(x))$ and $f(x)$ is $\Omega(g(x))$.

$a \mid b$ **(**a **divides** b**):** there is an integer c such that $b = ac$.

a **mod** b**:** the remainder when the integer a is divided by integer b.

$a \equiv b$ (**mod** m) (a **is congruent to** b **modulo** m)**:** $m \mid (a - b)$.

RESULTS

division algorithm: Let $a \in Z$ and $d \in Z^+$. Then there are unique $q, r \in Z$ with $0 \leq r < d$ such that $a = dq + r$.

INDUCTION & RECURSION

TERMS

the principle of mathematical induction: That the following statement is true:

$$(P(1) \wedge \forall k[P(k) \rightarrow P(k+1)]) \rightarrow \forall n P(n).$$

basis step: The proof of $P(1)$ in a proof by mathematical induction of $\forall n P(n)$.

inductive step: The proof of $\forall k[P(k) \rightarrow P(k+1)]$ in a proof by mathematical induction of $\forall n P(n)$.

strong induction: That the following statement is true:

$$(P(1) \wedge \forall k[(P(1) \wedge \cdots \wedge P(k)) \rightarrow P(k+1)]) \rightarrow \forall n P(n).$$

well-ordering property: Every nonempty set of nonnegative integers has a least element.

recursive definition of a function: a definition of a function that specifies an initial set of values and a rule for obtaining values of this function at integers from its values at smaller integers.

recursive definition of a set: a definition of a set that specifies an initial set of elements in the set and a rule for obtaining other elements from those in the set.

structural induction: a technique for proving results about recursively defined sets.

recursive algorithm: an algorithm that proceeds by reducing a problem to the same problem with smaller input.

COUNTING

TERMS

permutation: an ordered arrangement of the elements of a set

r**-permutation:** an ordered arrangement of r elements of a set

$P(n,r)$**:** the number of r-permutations of a set with n elements.

$C(n,r)$**:** the number of r-combinations of a set with n elements

$\left(\begin{array}{c} n \\ r \end{array} \right)$ **(binomial coefficient):** $C(n,r)$.

combinatorial proof of an identity: a proof that uses counting arguments to prove that both sides of an identity count the same set of objects in different ways

Pascal's triangle: a representation of the binomial coefficients where the ith row of the triangle contains $\left(\begin{array}{c} i \\ j \end{array} \right)$, for $j = 0, 1, 2, \ldots, i$.

RESULTS

product rule: a basic counting technique: the number of ways to do a procedure that consists of 2 subtasks is the number of ways to do the 1st subtask *times* the number of ways to do the 2nd subtask after the 1st subtask has been done

sum rule: a basic counting technique: the number of ways to do a task in 1 of 2 ways is the sum of the number of ways to do these tasks if they cannot be done simultaneously

pigeonhole principle: When more than k objects are placed in k boxes, there must be a box with more than 1 object.

generalized pigeonhole principle: When N objects are placed in k boxes, there must be a box with at least $\lceil N/k \rceil$ objects.

$P(n,r) = \frac{n!}{(n-r)!}$

$C(n,r) = \left(\begin{array}{c} n \\ r \end{array} \right) = \frac{n!}{r!(n-r)!}$

Pascal's Identity: $\left(\begin{array}{c} n+1 \\ k \end{array} \right) = \left(\begin{array}{c} n \\ k-1 \end{array} \right) + \left(\begin{array}{c} n \\ k \end{array} \right)$

Binomial Theorem: $(x+y)^n = \sum_{k=0}^{n} \left(\begin{array}{c} n \\ k \end{array} \right) x^{n-k} y^k$

There are n^r r-permutations of a set with n elements when repetition is allowed.

There are $C(n+r-1,r)$ r-combinations of a set with n elements when repetition is allowed.

There are $\frac{n!}{n_1!n_2!\cdots n_k!}$ permutations of n objects where there are n_i indistinguishable objects of type i, for $i = 1, 2, \ldots, k$.

Recurrence Relations

Terms

recurrence relation: a formula expressing terms of a sequence, except for some initial terms, as a function of 1 or more previous terms of the sequence

initial conditions of a recurrence relation: the values of the terms of a sequence satisfying the recurrence relation before this relation takes effect

divide-and-conquer algorithm: an algorithm that solves a problem recursively by splitting it into a fixed number of smaller problems of the same type

Results

$$|A \cup B| = |A| + |B| - |A \cap B|$$

Relations

Terms

binary relation from A to B: A subset of $A \times B$.

relation on A: a binary relation from A to itself.

$S \circ R$**:** $\{(s,t) \mid \exists x, (sRx \wedge xSt)\}$.

R^{-1}**:** $\{(t,s) \mid sRt\}$.

R^n**:** the n^{th} power of R.

reflexive: a relation R on A is *reflexive* if $\forall a \in A, aRa$.

symmetric: a relation R on A is *symmetric* if $\forall a, b \in A, aRb \rightarrow bRa$.

antisymmetric: a relation R on A is *antisymmetric* if $\forall a, b \in A, (aRb \wedge bRa) \rightarrow a = b$.

transitive: a relation R on A is *transitive* if $\forall a, b, c \in A, (aRb \wedge bRc) \rightarrow aRc$.

directed graph or digraph: a set of elements called *nodes* or *vertices* and ordered pairs of these elements, called *edges* or *arcs*.

path in a digraph: a sequence of arcs $(a, x_1), (x_1, x_2), \ldots, (x_n, b)$ such that the terminal node of each arc is the initial node of the succeeding arc in the sequence.

circuit (or cycle) in a digraph: a path in the digraph that begins and ends at the same node.

equivalence relation: a reflexive, symmetric, and transitive relation.

equivalent: if R is an equivalence relation, a is equivalent to b if aRb.

$[a]_R$ **(equivalence class of a with respect to R):** $\{b \mid aRb\}$.

partition of a set S: a collection of a pairwise disjoint nonempty subsets that have S as their union.

partial ordering: a relation that is reflexive, antisymmetric, and transitive.

poset (S, R): a set S and a partial ordering R on S.

comparable: the elements a and b in the poset $(A, \preceq)$ are *comparable* if $a \preceq b$ or $b \preceq a$.

incomparable: elements in a poset that are not comparable.

total (or linear) ordering: a partial ordering for which every pair of elements are comparable.

RESULTS

1. Let R be an equivalence relation. Then, the following 3 statements are equivalent:
   - aRb.
   - $[a]_R \cap [b]_R \neq \emptyset$.
   - $[a]_R = [b]_R$.
2. The equivalence classes of an equivalence relation on a set A form a partition of A. Conversely, an equivalence relation can be constructed from any partition so that the equivalence classes are the subsets in the partition.

GRAPHS

TERMS

undirected edge: An edge associated with a set $\{u, v\}$, where u and v are vertices.

directed edge: An edge associated with an ordered pair (u, v), where u and v are vertices.

loop: An edge connecting a vertex with itself.

undirected graph: A set of vertices and a set of undirected edges each of which is associated with a set of 1 or 2 of these vertices.

simple graph: An undirected graph with no multiple edges and no loops.

multigraph: An undirected graph that may contain multiple edges but no loops.

directed graph: A set of vertices and a set of directed edges each of which is associated with an ordered pair of vertices.

adjacent: Two vertices are adjacent if there is an edge between them.

incident: An edge is incident to a vertex if the vertex is an endpoint of that edge.

$deg(v)$ **(the degree of the vertex v in an undirected graph):** The number of edges incident to v with loops counted twice.

$deg^-(v)$ **(the in-degree of the vertex v in a graph with directed edges):** The number of edges with v as their terminal vertex.

$deg^+(v)$ **(the out-degree of the vertex v in a graph with directed edges):** The number of edges with v as their initial vertex.

K_n **(Complete graph on n vertices):** The undirected graph with n vertices where each pair of vertices is connected by an edge.

bipartite graph: A graph with a vertex set that can be partitioned into subsets V_1 and V_2 such that each edge connects a vertex in V_1 and a vertex in V_2.

$K_{m,n}$ **(Complete bipartite graph:** The graph with a vertex set partitioned into a subset of m vertices and a subset of n vertices such that 2 vertices are connected by an edge if and only if one vertex is in the first subset and the other is in the second subset.

C_n **(cycle of size n), $n \geq 3$:** The graph with n vertices $v_1, v_2, \ldots, v_n$ and edges $\{v_1, v_2\}$, $\{v_2, v_3\}$, $\ldots$, $\{v_{n-1}, v_n\}$.

adjacency matrix: A matrix representing a graph using the adjacency of vertices.

incidence matrix: A matrix representing a graph using the incidence of edges of vertices.

circuit: A path of length $n \geq 1$ that begins and ends at the same vertex.

connected graph: An undirected graph with the property that there is a path between every pair of vertices.

strongly connected directed graph: An directed graph with the property that there is a directed path from every vertex to every vertex.

Euler circuit: A circuit that contains every edge of the graph exactly once.

Hamilton circuit: A circuit in a simple graph that visits each vertex exactly once.

RESULTS

1. There is an Euler circuit in a connected multigraph if and only if every vertex has even degree.