|  |  |  |  |
| --- | --- | --- | --- |
| Propositional Logic  A proposition is a declarative sentence that is either T (1)/F (0)     * Negation * Conjunction * Disjunction * Implication, if-then   T for everything else except for     * Biconditional, if-and-only-if   T when both has the same value (T/F)  **Tautology:** a proposition that is always true  **Contradiction:** a proposition that is always false   * **Contrapositive** of is for      * **Converse** of is for   is NOT LOGICALLY EQUIVALENT to   * **Inverse** of is for   is NOT LOGICALLY EQUIVALENT to  **De Morgan’s Law**    **Predicates & Quantifiers**   * Predicate: A statement whose truth value depends on one or more variables. * **Universal Quantifier, :** “For all x, P(x) is true” 🡪 * **Existential Quantifier**, : “There exist some x, where P(x) is true” 🡪 * Image result for xor truth table**Domain of discourse:** Object/Category/Item being considered   Proof   1. Direct proof,   If p = T, then q = T   1. Contrapositive, 2. Vacuous proof,   If p = F, then q can be T or F to be T.   1. Trivial proof,   If q = T, then p can be T or F to be T.   1. Proof by Equivalence,   Show that if | 1. Proof by Contradiction: Negate the current statement and assume that the negated statement is T until proven F. 2. Proof by Counterexample: Show that at least in one condition, the statement is F. 3. Constructive proof: For proving . Proof P(x) = T for 1 element in the domain discourse. 4. Proof by Cases: Identify all possible sub-cases and prove.       Rules of Inference  Premises (assume true)  Conclusion  (true)  Set theory     1. **A set** {}  * Repetitions removed, ex: if A = {1, 1, 2, 3}, write as A = {1, 2, 3} * Order of elements in a set doesn’t matter, ex {1, 2, 3} = {2, 3, 1} * Is a subset of itself, * The empty set is a subset of every set, | | |
| **Operation on sets:**   1. Union: 2. Intersection: 3. Complement: 4. Set difference:   **Subsets vs set membership vs equality**   1. - TRUE 2. – TRUE 3. – TRUE 4. – TRUE, sets of “sets” 5. – WRONG 6. – WRONG 7. **Ordered pair, tuple ()**   Order matters,  **Cartesian Product**   * Cartesian product of set A and B is the set of all ORDERED PAIRS (a, b),   **Cardinality of sets (Size)**   * Cardinality of * **Power set** of A is the set of all subsets of A, if A = {1, 2,3}, then the power set:   Functions   * X – domain of f, Y – co-domain of f. * **,** x pre-image of y, y is the image of x under f. * Not every element of Y necessarily gets a pre-image. * If are images of x under f, it is called the range of f, Range(f)  1. **Injective Function** One-to-one  * **Each element in the co-domain Y** has **AT MOST 1** incoming arrow. * To prove: Do **proof-by-contradiction**. Assume that but   Deduce that , thus leading to a contradiction.   * To disapprove: Do **proof-by-counterexample.** Find such that . For onto, find  1. **Surjective Function** Onto  * **Each element in the co-domain Y** has **AT LEAST 1** incoming arrow. * To prove: Do **direct proof.** Assume some generic y in co-domain and prove that for some x in domain. * To disapprove: Do **counterexample**. Find y that has no inverse image in X.  1. **Bijection Function** 1-to-1 and onto  * **One arrow out of every element in domain X** * **One arrow into every element in co-domain Y** | | 1. **Composition of function**     Relations   * where R is a relation that makes * Every function is a relation BUT NOT vice versa.  1. **Properties of Relation** 2. **Reflexivity:** R is reflexive if aRa 3. **Symmetry:** R is symmetric if aRb iff bRa 4. **Transitivity:** R is transitive if for any triple aRb and bRc, then aRc. 5. **Irreflexive:** R is irreflexive if for all element a of A is not related to itself. 6. **Antisymmetric:** R is antisymmetric if I. If aRb and bRa, then a = b II. Exist only aRb, no bRa 7. **“Equivalence” Relation:**  * Reflexive   If R is an equivalence relation: if aRb, then a is equivalent to b.   * Transitive * Symmetric  1. **Equivalence Classes**      1. **Partitions**  * Grouping of elements of a given set **into DISJOINT SUBSETS.** * The union of subsets gives us the WHOLE set. * The subsets are DISJOINT for ALL PAIR of subsets.  1. **Order relations** (can use Hasse Diagram)  |  |  | | --- | --- | | **Partial order** | **Total order** | | * Reflexive * Transitive * Antisymmetric   \*Not every pair of elements need to be related | * For either * Antisymmetric * Transitive   \*Any total order is also a partial order. | |
| Graph  **Directed Graph**   * Vertex degree: in-degree and out-degree * E = (u, v), u – start vertex, v – end vertex   **Undirected Graph**   * The relation is symmetric for the pairs. * Undirected edge between nodes a and b, {a, b} since order doesn’t matter * Vertex degree: number of edges touching that node. * Vertices and b are ADJACENT if they are connected by an edge. * Given E = (a, b), then E is said to be an incident on a and b.   **Simple Graph:**   * Undirected graph * No self-loops * No multiple edges between nodes. At most one edge between distinct vertices.   **Representing graphs:**    **Degree theorem, for undirected graph:**  **Handshaking Lemma:**  **Degree theorem, for directed graph:**  **Graph Traversal**   * Walk – Both edges and nodes can be visited more than once. * Path – any walk that does not contain repeated edges. However, nodes can be repeated. * Simple path – any path that does not contain repeated vertices.   **Bipartite Graphs,**   * m vertices on one side and n vertices on another. Edges = m \* n * **Hall’s Marriage Theorem: A** bipartite graph, G = has a complete matching from iff if for all subsets . | | **Special Graphs**   * Line graph. [ ](n nodes, n-1 edges) * Cycle, [Closed loop] () * Tree, [Connected graph containing no cycles] * Star graph, [central node connected to the other outer nodes] (n nodes, n-1 edges) * Wheel, [star graph but the outer is connected] (n nodes, 2(n-1) edges) * Complete graph, , [connecting each of the n nodes with every other node] (n nodes, edges)   **Connectivity**   * **Strongly connected graph:** It is possible to reach every vertex from every other vertex for all cases. * **Weakly connected graph:** The underlying graph is fully connected.   Mathematical Induction (Base case & Inductive Step)   * Provide: Predicate, Base case, Inductive step   **Inductive Step**   * **General form of Mathematical Induction**: **Assume P(k) is true and prove P(k+1) is true.** * Suppose P(k) is true for [ ex: k > 0, equation] * show that P(k+1) is true by making use of P(k) since P(k) is supposed to be true. * **Strong Induction: Assume that all of P(1), P(2),..,P(k) is true and use any combination of these k predicates to prove P(k+1) is true.** * Use when the truth of P(k+1) requires that P(I) is true for ALL integers * Necessary to prove multiple base cases   **Arithmetic Progressions**    **Geometric Progressions**    **Summations and Recursions** |
| **Time Complexity**  Data Structure: Time complexity chart of searching and sorting  Counting  1. **Way to choose an element**   1. **Addition Rule**  * Disjoint sets:   then (cardinality/size)   * Non-disjoint sets, PIE:  1. **Product Rule**  * is the Cartesian product of A and B, containing all ordered pairs (a, b) where . * Number of possibilities (a, b): * If there are different conditions (ex: password length 7, 8, 9) **I. Find the possibilities of 7, 8, 9 II. Sum it all.**   2. **Permutations and Arrangement Principle** (order of selection matters)   * The number of ways to arrange r distinct elements drawn from a set of n elements:   n = total num. of elements, r = num. of place (\_ \_ \_)   * The number of ways to arrange all n elements of the set into a sequence: * Hence, to find the number of possible ways with different arrangements:   3. **Combinations and Selection Principle** (the order of selection doesn’t matter, no repeats)   1. **Division Rule**  * Used when we want permutation without repeats. Ex: the knight example = (a, b, c), (b, c, a), (c, a, b) – DON’T WANT REPEATS * “n choose r”:   4. **Counting with Repetitions** (Ball & Sticks argument)   * Donut examples: 17 donut, 4 varieties * n objects/item from r varieties:   5. **Pigeon Hole**  If N pigeons are placed in K holes, then there is at least A HOLE that holds at least   * How many different functions are there from a set with 10 elements to sets with 2 elements? Answer:   How many different functions are there from a set with 10 elements to sets with 5 elements? Answer: | |  |