**Universal statement**: property is true for all elements of a set

**Conditional statement**: if one thing is true some other thing also has to be true

**Existential statement**: there is at least one element for which the property is true

**Universal existential statement**: First part: property is true for all, second part: something exists

**Existential universal statement**: First part: asserts the existence of an object, second part: object satisfies property for all things of a certain kind

|  |  |
| --- | --- |
| *x ∈ S* | x is an element of set S. Same element multiple times counts as one, order is irrelevant |
| *{x ∈ S | P(x)}* | The set of all elements such that the property P(x) is true for all x in S |
| *A ⊆ B* | A is a subset of B. If x is an element of A, then x is also an element of B. |
| Proper subset | All x in A are in B, but there is at least one y in B that is not in A. |
| (a,b) | Ordered pair. Position is relevant |
| A × B | Cartesian product. Set of ordered pairs with all a*∈*A and b*∈*B (a,b) |
| A R B | Subset of A × B. Relation from A to B. Some ordered pairs of A × B, with (x,y) *∈* R. |
| (x,y) *∈* F | Function. For every element x in A there is an element y in B. For every x, there is only one y, but one y can have multiple x. y=F(x) |
| Left-unique | For every y there is exactly one x |
| Right-unique | For every x there is exactly one y |
| Left-total | For every x there is a y |
| Right-total | For every y there is a x |
| ∼ | Not, negation |
| ∧ | And, conjunction |
| ∨ | Or, disjunction |
| Tautology | Always true, regardless of truth values |
| Contradiction | Always false, regardless of truth values |
|  | |
| p→q | “p implies q”, is false when p is true and q is false |
| p↔q | Biconditional, false if p and q have different truth values |
| ∴ | “therefor”, conclusion |
| Rules of Inference:   * Generalization: If p is true, then p or q is true for any other statement q * Specialization: Discard extraneous information * Elimination: If “p or q” and q is false, then p must be true * Transitivity: If p→q and q→r then p→r * Division into cases: If either p or q has to be true and “p or q” and p→r and q→r, then r is true | |
| Fallacies:   * Using ambiguous premises * Circular reasoning, assuming what is to be proved without having derived it from the premises * Jumping to a conclusion without adequate grounds | |
| Converse error | Assuming p↔q when only p→q and assuming p is true when q is true |
| Inverse error | Assuming ∼p→∼q when only p→q is given |
|  | |
| Recognizer | Circuit that outputs true for exactly one particular combinaton of input signals and false for all others |
| P|Q | ∼(P∧Q) |
| PQ | ∼(P∨Q) |
| Predicate | Sentence that contains variables. Becomes a statement if values are put in place of the variables. |
| Domain | Set of all values that may be substituted in place of the variable |
| Truth set | Set of all elements D that make the predicate P(x) true when substituted for x. Denoted as |
| ∀ | For all |
| ∃ | There exists |
| 𝐵 ⊇ A | Superset, *A ⊆ B* |
| (a,b) |  |
| [a,b] |  |
| (a,b] |  |
|  |  |
| Disjoint | No members in common |
| Partition of a set | Mutually disjoint subsets of the main set and all subsets combined being equal to the main set. |
| |A| | Cardinality. Number of elements of A if A is finite |
| Range of function f |  |
| Injective | Function where every element in the co-domain is only with one x |
| Surjective | Every y is connected to at least one x. |
| Bijective | Injective and surjective |
| Inverse function | Function from y to x that undoes the function from x to y. Only possible with bijective functions |
|  | Composition of functions |
|  | Identity function. Function from X to X. |
| Cardinality | Number of elements in a set |
| x R x for all x in X | Reflexive relation |
| x R y then y R x | Symmetric relation |
| X R y and y R z, then x R z | Transitive relation |
| Equivalence relation | Reflexive, symmetric and transitive relation |
| Rt | adding the least number of ordered pairs to ensure transitivity is called the transitive closure of the relation |
| Partition of A | Collection of mutually disjoint subsets of A, whose union is A |
| M = n(mod d) | M is congruent to n modulo d |
| Vertices | Dots of a graph |
| Edges | Lines in a graph, connect vertices |
| Edge-endpoint function |  |
| Parallel edges | Edges that connect the same vertices |
| Adjacent edges | Edges that have 1 vertex in common |
| Isolated vertex | Vertex without any edges |
| Adjacent vertices | Vertices that are connected through an edge |
| Directed edge | Arrow, written as an ordered pair of vertices instead of set |
| Complete graph | All pair combinations of vertices are connected by edges |
| Subgraph | Graph that contains parts of the main graph |
| Degree of vertex | Number of edges of the vertex. Loop count as 2 |
| Total degree of graph | 2x number of edges |
| Trails | Routes that do not have a repeated edge |
| Paths | Trails that do not have repeated vertices |
| Circuits | Trails that start and end at the same vertex |
| Simple circuits | Circuits that only pass through each vertex only once, except the start vertex |
| Connected graph | Possible to travel from any vertex to any other vertex along adjacent edges |
| Connected component | Subgraph that is connected |
| Euler circuit | circuit that passes through all the vertices of G at least once and passes through all the edges only once |
| Hamilton circuit | Circuit that passes through all the vertices only once except for start |
| Matrices and directed graphs |  |
| Matrices and undirected graphs |  |
| Matrices and connected components |  |
| Isomorphisms of graphs | If the structure of a graph stays the same, but the vertices and edges are labeled differently |
| Tree | Connected graph, that does not contain circuits |
| Rooted tree | Tree with one vertex, from which a path to every other vertex is possible |
| Octree | Tree, where each internal vertex has eight children |
| Spanning tree | Subgraph of G that contains every vertex of G and is a tree |
| Weighted graph | A graph whose edges are labeled with numbers (weights) |
| minimum-weight spanning tree,  minimum spanning tree | spanning tree for which the sum of the weights of all the edges is as small as possible |
| Kruskal | Start with minimum weight. Add next smallest unless it makes a circle. Fast in sparse graphs |
| Prim | Start with random vertex. Add vertices so that they are connected. Fast in dense graphs |
|  |  |
|  |  |
|  |  |