Glossary for CMIS 160

(Revised 26 January 2005)

Part I: Numbers

Term	Applies to	Meaning	Remarks
composite	An integer	An integer x is composite if it can be written as a product $x = ab$ where neither a nor b is any of 0, 1, or –1.	Note that under this definition, - 4 is composite, since it can be written as 4 = (2)(-2). This is the accepted definition from advanced mathematics. Some authors, including Epp, do not apply the terms "prime" and "composite" to negative integers.
congruent	A pair of integers	a is congruent to b mod n, written $a \equiv b \pmod{n}$, if there is an integer k such that $a - b = kn$.	Equivalently, $n \mid (a - b)$. See *divides*.
divides	A pair of integers	a divides b, written $a \mid b$, if there is an integer k such that $b = ka$.	
even	An integer	An integer x is even if there is an integer y such that $x = 2y$.	
irrational	A real number	Not rational.	Examples of irrational numbers include $\sqrt{2}$, $\log_2 5$, and π.
odd	An integer.	An integer x is odd if there is an integer y such that $x = 2y + 1$.	Equivalently, x is not even..
prime	An integer	An integer x is prime if (a) x is not 0, 1, or –1, and (b) if x is written as a product $x = ab$, then either a or b is equal to either 1 or –1.	Note that under this definition, -3 is prime. This is the accepted definition from advanced mathematics. Some authors, including Epp, do not apply the terms prime or composite to negative integers.
rational	A real number	A real number x is rational if there are integers a and b such that $x = a/b$.	Not all real numbers are rational. See *irrational*.

Part II: Logic, Sets, Functions, and Relations

Term	Applies to	Meaning	Remarks
bijection	A function	A function that is one-to-one and onto.	See *one-to-one correspondence*.
bijective	A function	One-to-one and onto.	
co-domain	Of a function	The set of values that the function is *allowed* to take.	Also called the *range* of the function. Unfortunately, the term *range* is ambiguous, and the term co-domain is rarely used. See *range* and *image*.
complement	Of a subset of a set	If $A \subset B$, then the complement of *A* in *B* is the set of elements of *B* that are not in *A*.	The complement of *A* is sometimes written A^c – but only in contexts wherein it is obvious what the set *B* is! The complement of *A* in *B* can always be written $B - A$. See *difference*.
contrapositive	Of a conditional statement.	The *contrapositive* of $p \Rightarrow q$ is the statement $\neg q \Rightarrow \neg p$.	The contrapositive of a conditional statement is logically equivalent to the original statement.
converse	Of a conditional statement.	The *converse* of $p \Rightarrow q$ is the statement $q \Rightarrow p$.	Either a statement or its converse can be true while the other is false!
Cartesian product	Of two sets	The Cartesian product of *A* and *B* is the set of all ordered pairs (a,b) where $a \in A$ and $b \in B$.	The Cartesian product of A and B is written $A \times B$. If *A*=*B*=**R** (where **R** is the set of real numbers), then $A \times B$ is the Euclidean plane, with the familiar Cartesian coordinates.
difference	Of two sets	The difference of *B* in *A*, written $A - B$, is the set of those elements of *A* that are not in *B*.	This is defined whether B is a subset of *A* or not. Contrast *complement.*
domain	Of a function.	The domain of a function *f* is the set of values *x* for which *f(x)* is defined.	If *f* is a function with domain *A* and co-domain *B*, we write $f : A \rightarrow B$
element	Of a set	An object that is a member of the set.	If a is an element of X, we write $a \in X$.
empty set		The set containing no elements at all.	It is denoted by $\varnothing$ or (less commonly) by φ.

Term	Applies to	Meaning	Remarks
equivalence relation	A type of relation	A relation that is reflexive, symmetric, and transitive.	An equivalence relation on a set divides the set up into *equivalence classes*. All members of an equivalence class are related to one another, but not to any member of any other equivalence class.
function		A rule or procedure that assigns to each element of a set A, a unique element of a set B.	If $x \in A$, then the result of applying the rule or procedure to x is denoted by $f(x)$.
identity	In a number system, relative to an operation.	An identity for an operation is an element that leaves all other elements unchanged under the operation.	0 is an identity relative to addition, since $(\forall x)(0+x=x)$. 1 is an identity for multiplication. The empty set $\varnothing$ is an identity under the set operation *union*.
identity function		A function on a set that assigns to each element of the set, itself.	The identify function on a set A has domain A and co-domain A ($f: A \to A$). If f is the identity function on A, then $f(x) = x$ for all $x \in A$.
image	Of a function	The set of all values *actually assumed* by the function.	If $f: A \to B$, then the image of f is $\{y \in B \mid (\exists x \in A)(f(x)=y\}$
injective	A function	Same as *one-to-one*.	
intersection	Two or more sets	The set consisting of all the elements that belong to all of the sets.	Denoted by $\cap$, such as $A \cap B \cap C$
inverse	Of a conditional statement.	The *inverse* of $p \Rightarrow q$ is the statement $\neg p \Rightarrow \neg q$.	Either a statement or its inverse can be true while the other is false!
inverse	Of a function	The inverse of a function f is defined if and only if f is one-to-one and onto (that is, f is a bijection). If $f: A \to B$, then the inverse g of f is that unique function $g: B \to A$ for which $g(f(x)) = x$. Also, $f(g(y)) = y$.	

Term	Applies to	Meaning	Remarks
inverse image	Of a subset of the co-domain of a function	The set of elements x in the domain for which *f(x)* is in the specified subset.	The inverse image of a set Y under the function f is denoted by $f^{-1}(Y)$. If $f : A \to B$, and if $Y \subset B$, then $f^{-1}(Y) = \{x \in A \mid f(x) \in Y\}$.
negation	Of a statement	The negation of a statement p is the assertion that p is false.	The negation of p is denoted by either $\sim p$ or $\neg p$. The two notations appear to be equally common in the literature.
one-to-one	A function	A function is one-to-one if it takes different values for different values of its argument.	A function $f : A \to B$ is one-to-one if $(\forall x, y \in A)(f(x) = f(y) \Rightarrow x = y)$
one-to-one correspondence	A function	Same *as bijection*.	The term sometimes causes confusion, since a one-to-one correspondence is not the same as a function that is one-to-one.
onto	A function	The function actually assumes all possible values; that is, its image is equal to its co-domain.	A function $f : A \to B$ is onto if $(\forall y \in B)(\exists x \in A)(f(x) = y)$. Note that "onto" is used as an adjective, not as a preposition!
power set	A set	The set of all subsets of the set.	If a set A has n elements, then its power set has 2^n elements.
predicate		A function whose co-domain is the set {TRUE, FALSE}. More informally, it is a statement about members of a set which is either true or false for each member of the set.	
proper subset	Of a set	A subset that does not contain all the elements of the set of which it is a subset.	There is no standard symbol for "proper subset". Some authors who use the symbol $\subseteq$ to mean "is a subset of" use the symbol $\subset$ to mean "is a proper subset of". See *subset*.

Term	Applies to	Meaning	Remarks
range	Of a function	(1) The set of values that the function *may* assume. (2) The set of values that the function *does* assume.	*The bad news*: This term is ambiguous, and is used in the literature with the meanings (1) *co-domain* and (2) *image*. You have to figure out from the context what the author intends. Epp uses "range" to mean "image" while most other authors use "range" to mean " co-domain". *The good news*: 98% of the time, the distinction is irrelevant to the practical problem at hand, so you don't have to worry about it at all.
reflexive	A relation on a single set A (i.e relating A to A.)	A relation is reflexive if every element is related to itself. I.e., $(\forall x \in A)xRx$.	
relation		A relation relating a set A to a set B can be thought of in either of two equivalent ways: either (a) a subset of the Cartesian product set $A \times B$, or as a predicate on $A \times B$.	If R is a relation relating A to B, and if $(x, y) \in R$ (or equivalently, $R(x, y)$ is TRUE) then we write *a R b*.
set		A collection of objects. This is an intuitive term.	
subset	Of a set.	A is a subset of B if every element of A is also an element of B.	The symbol $\subset$ is almost universally used to mean "is a subset of". Some authors (including Epp) use the symbol $\subseteq$ instead. See *proper subset*.
superset	Of a set.	The opposite of *subset*. A is a superset of B if and only if B is a subset of A.	The symbol $\supset$ is almost universally used to mean "is a superset of". Some authors (including Epp) use the symbol $\supseteq$ instead.
surjective	A function.	Same as *onto*.	

Term	Applies to	Meaning	Remarks
symmetric	A relation on a single set A (i.e relating A to A.)	Such a relation is symmetric if $(\forall x \in A)(\forall y \in A)(xRy \Rightarrow yRx)$.	
tautology	A formula of prepositional calculus	A formula is a tautology if and only if it takes the value TRUE for all possible combinations of truth-values for the variables in the formula.	An argument form "If A then B" is valid if and only if $A \Rightarrow B$ is a tautology.
transitive	A relation on a single set A (i.e relating A to A.)	Such a relation is transitive if $(\forall x, y, z \in A)(xRy \wedge yRz \Rightarrow xRz)$.	
union	Two or more sets	The set consisting of all the elements that belong to one or more of the sets.	Denoted by $\cup$, such as $A \cup B \cup C$

Part III: Graphs and Trees

Term	Applies to	Meaning	Remarks
adjacent	Two vertices of a graph	There is an edge connecting the two vertices.	
adjacent	Two edges of a graph	There is a vertex that is common to the two edges	
ancestor	Two vertices in a rooted tree.	Vertex *w* is an ancestor of *v* if *v* is a descendent of *w*.	The ancestors of *v* are those vertices that lie on the path from *v* to the root (excluding *v* itself, of course). The root is an ancestor of every other vertex.
binary tree		A rooted tree in which (a) every parent has at most two children, (b) each child is designated as either a right child or a left child, and (c) if a parent has two children, then one is left and one is right.	
branch vertex	Of a tree	Same as internal vertex.	Rarely used.
child	Of a vertex in a rooted tree	Vertex *v* is a child of vertex *w* if *v* is adjacent to *w* and is further from the root than *w*.	In this case, *w* is called the parent of *v*. The level of *v* is one more than that of *w*.
circuit	In a graph	A closed walk that is a path.	I.e., it does not contain any repeated edge
circuit-free	A graph	A graph is circuit free if it has no non-trivial circuits.	We have to say "non-trivial" since a trivial walk (a single vertex) is, strictly speaking, a circuit. A trivial circuit has no repeated edges because it has no edges at all!
closed walk	In a graph	A walk that starts and ends at the same vertex.	

Term	Applies to	Meaning	Remarks
complete bipartite graph on (m,n) edges	Positive integers *m* and *n*.	A graph such that the set of vertices is partitioned into a subset V_1 having *m* vertices, and a subset V_2 having *n* vertices, and such that for every vertex v_1 in V_1 and for every vertex v_2 in V_2, there is exactly one edge connecting v_1 to v_2 – and no other edges.	Denoted by $K_{m,n}$. $K_{m,n}$ has *mn* edges. There are no edges connecting a vertex of V_1 to another vertex of V_1, nor any edges that connect a vertex of V_2 to any other vertex of V_2.
complete graph on n vertices	A positive integer *n*.	A graph such that for each pair of (distinct) vertices, there is exactly one edge connecting them – and no other edges	Denoted by K_n. K_n has C(*n*,2) edges.
connect	Two vertices and an edge.	The vertices are those assigned to the edge by the edge-endpoint function.	
connected	Two vertices in a graph	Two vertices are connected if there is a walk from one to the other.	In fact, if two vertices are connected, you can find a path from one to the other, by eliminating any backtracking or looping in the walk.
connected	A graph	A graph is connected if every vertex is connected to every other vertex.	
connected component	Of a graph	A maximal connected subgraph.	If a graph is not connected, it is made up of two or more connected components. An isolated vertex is a connected component all by itself.
degree	Of a vertex	The number of edges incident to the vertex, with each loop counted twice.	
descendant	Two vertices in a rooted tree.	Vertex *v* is a descendent of vertex *w* if $w \neq v$ and the path from *v* to the root passes through *w*.	Every vertex other than the root is a descendent of the root.
directed graph		A structure consisting a set of vertices, a set of edges, and a function that assigns to each edge an ordered pair of vertices.	Equivalently, you can think of a directed graph as a set of vertices *V*, a set of edges *E*, and two functions from *E* to *V*: the *initial vertex function* and the *terminal vertex function*.

Term	Applies to	Meaning	Remarks
distance	Between two vertices of a graph	The length of the shortest path connecting the vertices.	If the graph is not connected, the distance between vertices in different connected components is considered to be infinite.
edge-endpoint function	A graph	The function that assigns one or two vertices to each edge of a graph.	
endpoint	An edge of a graph	The vertex, or either of the two vertices, that are assigned to an edge by the edge-endpoint function of the graph.	
Euler circuit	In a graph	A circuit that contains every vertex and every edge.	An Euler circuit is sometimes incorrectly defined as "a circuit that contains every edge"; but that definition would permit isolated vertices.
Euler path	In a graph	A path, connecting two distinct vertices, that contains every edge of the graph.	In a connected graph, an Euler path from v to w exists if and only v and w are of odd degree, and are the only vertices in the graph to have odd degree.
forest	A graph	A graph that is circuit-free.	If a forest is connected, it is a tree. The connected components of a forest are trees.
full binary tree		A binary tree in which each parent has exactly two children.	
graph	A function	If f: $X \to Y$ is a function, then the graph of the function f is the set of all ordered pairs (x,y) in the Cartesian product $X \times Y$ such that $y = f(x)$.	This is a special case of the graph of a relation.
graph	A relation	If R is a relation relating the sets X and Y, then the graph of R is the set of all ordered pairs (x,y) in the Cartesian product $X \times Y$ such that $x \, R \, y$.	This generalizes the concept of the graph of a function.

Term	Applies to	Meaning	Remarks
graph	Neither a function nor a relation.	A structure consisting of a set of vertices, a set of edges, and a function (the "edge-endpoint function") that assigns to each edge a set consisting of either one or two vertices.	This usage has nothing whatsoever to do with graphs of functions or relations. You should be aware that some authors use the term "graph" to mean what we call a "simple graph". These authors use terms like "multigraph" or "pseudograph" to mean what we call a graph.
Hamiltonian circuit	In a graph	A simple circuit that contains every vertex.	There is no easy way to tell whether a Hamiltonian circuit exists in a graph.
height	Of a rooted tree	The maximum of the levels of all the vertices.	
incident	An edge and a vertex	The vertex is an endpoint of the edge	
internal vertex	Of a tree	A vertex that is not a leaf.	Also called branch vertex.
isolated vertex	Of a graph	A vertex that is not an endpoint of any edge.	
isomorphic	Two graphs	Graphs G and H are isomorphic if there is an isomorphism from G to H.	
Isomorphic invariant	Property of a graph	A property of a graph that is shared by every graph that is isomorphic to it.	
isomorphism	Two graphs	An isomorphism f from graph G to graph H consists of two functions $f_V: V(G) \rightarrow V(H)$ and $f_E: E(G) \rightarrow E(H)$ such that (a) both of these functions are bijections, and (b) they preserve the edge-endpoint function.	This is a fancy way of saying that you can make G look just like H by renaming the vertices and edges.
leaf	Of a tree	A vertex (of a tree) that has degree zero or one.	Same as terminal vertex. The term leaf is the more common. The only case in which a leaf has degree zero is when the graph is the trivial tree consisting of a single vertex and no edges.

Term	**Applies to**	**Meaning**	**Remarks**
left child	Vertex in a rooted tree	See binary tree.	
left subtree	A vertex in a binary tree	The left subtree of a vertex *v* is the connected subgraph of the tree containing the left subchild of *v* and all of its descendents.	The left subtree is a rooted tree, with the left child of *v* being its root.
length	Of a walk	The number of edges in the walk.	May be zero (in the case of the a trivial walk).
level	Of a vertex in a rooted tree.	The distance from the vertex to the root.	
loop	An edge in a graph	An edge is a loop if it has only one vertex assigned to it by the edge-endpoint function.	
minimal spanning tree	Of a weighted tree	A spanning tree of a weighted graph that has the smallest total weight (sum of the weights of its edges) of all spanning trees of the graph	A weighted tree always has one, but it need not be unique. Kruskal's Algorithm and Prim's Algorithm are efficient methods for finding minimal spanning trees.
parallel	Two edges in a graph	Two edges are parallel if they have the same sets of endpoints.	
parent	Of a vertex in a rooted tree	Vertex *w* is a parent of vertex *v* if *v* is a child of *w*.	Every vertex of a rooted tree, except the root, has exactly one parent.
path	In a graph	A walk with no repeated edges	A path may have repeated vertices, however.
right child	Vertex in a rooted tree	See binary tree.	
right subtree	A vertex in a binary tree	The right subtree of a vertex *v* is the connected subgraph of the tree containing the right subchild of *v* and all of its descendents.	The right subtree is a rooted tree, with the right child of *v* being its root.
root	Of a rooted tree	The vertex that is identified as the root.	
rooted tree		A tree in which one vertex is identified as the root.	

Term	Applies to	Meaning	Remarks
sibling	Of a vertex in a rooted tree	A sibling of *v* is a vertex, other than *v*, having the same parent as *v*.	As stated here, a child is not a sibling of itself, that is, the sibling relation is not reflexive. The literature is ambiguous on this point.
simple circuit	In a graph	A circuit that does not contain any repeated vertex.	It is tempting to say that a simple circuit is a closed walk that is a simple path. But this is not quite true: a closed walk always contains a repeated vertex (its start/end point) whereas a simple path contains no repeated vertex whatsoever.
simple graph		A graph without loops or parallel edges.	
simple path	In a graph	A path without repeated vertices.	A simple path can never contain a loop, since that would result in a repeated vertex. (Not containing a loop is a necessary, but not a sufficient, condition for a path to be simple.)
spanning tree	Of a graph	A subgraph of the graph that (a) contains all of the vertices of the graph, and (b) is a tree.	Every connected graph has at least one spanning tree.
subgraph	Of a graph	H is a subgraph of G if (a) every vertex of H is a vertex of G, (b) every edge of H is an edge of G, and (c) the endpoints of each edge of H are the same as they are in G.	
terminal vertex	Of a tree	A vertex (of a tree) that has degree zero or one.	Also called a leaf. We permit degree zero so that if the tree is just a single (isolated) vertex, that vertex is a terminal vertex.
total degree	Of a graph	The sum of the degrees of all the vertices.	The total degree of a graph is always even, being equal to twice the number of edges.
tree	A graph	A connected graph that is circuit-free.	

Term	Applies to	Meaning	Remarks
trivial walk	A graph	A walk consisting of a single vertex (and therefore, no edges).	A trivial walk has length zero.
vertex	A graph	One of the "points" or other objects that are connected by the edges of the graph.	
walk	In a graph	An alternating sequence of vertices and edges of a graph that (a) the sequence both starts and ends with a vertex, and (b) each edge connects the two vertices that it is between.	
weighted graph	A graph	A graph is which a numerical value is associated with each edge. The values are called the weights of the edges.	In most practical applications, the weights are positive, but this is not specified in the definition.

Part IV: Formal Languages and Finite State Automata

Term	Applies to	Meaning	Remarks
accepting state	A finite state automaton	If a string is fed into a FSA, the string is "accepted" by the FSA if and only if the FSA halts in an accepting state.	
alphabet		A set of symbols	Always non-empty
concatenation	Of two strings	Forming a new string by joining the two strings end to end.	
context-free language		A class of languages that includes almost all programming languages and almost all natural languages	Context-free languages are those defined by context-free grammars
finite-state automaton (FSA)		A machine defined by an input alphabet, a set of states, an initial state, and a set of accepting states	Also called *finite-state machine*. Some authors omit the initial state and the set of accepting states from the definition
finite-state machine (FSM)		Same as *finite-state automaton*.	
formal language		A set of strings over an alphabet	
grammar		A way of defining a formal language, by specifying rules for constructing the strings that belong to the language	
Initial state	An FSA	A state of a FSA, at which it starts operation.	
Input alphabet	An FSA	The set of symbols that are allowed in the input string provided to the FSA	
Kleene closure	An alphabet	The set of all strings over the alphabet	Includes the empty string
Kleene closure	A language	The language consisting of all strings that can be formed by concatenating strings of the original language	
language		See *formal language*	
next-state function	An FSA	The function that defines the next state of the FSA, given the current state and the input symbol	

Term	Applies to	Meaning	Remarks
next-state table	An FSA	A table that defines the next-state function of an FSA	Usually annotated to show the initial state and the accepting state
non-terminal symbol	A grammar	A symbol that is not allowed to appear in the language defined by the grammar	
positive closure	An alphabet	The set of all strings over the alphabet, except for the empty string	
production	A grammar	A rule stating what strings may be substituted for what symbols in building sentences of the language	
regular expression		A character string, allowed to contain the symbols of an alphabet and also the characters ( ) * and |, that defines a language	There are rules for correct formation of regular expressions. Regular expressions define regular languages.
regular grammar		A grammar in which all allowed productions are of the form $X \leftarrow x_1x_2\ldots x_nY$, where X is a non-terminal symbol, Y is a non-terminal symbol or absent, and the x_i are symbols in the grammar's alphabet.	Regular grammars define regular languages.
regular language		(a) A language that can be defined by a regular expression. (b) A language that can be defined by a regular grammar. (c) A language accepted by an FSA.	All three characterizations are equivalent. (This face is known as Kleene's Theorem.)
state	An FSA	An FSA has a set of states. At any point during operation, the FSA is in one state.	
string	Over an alphabet	A finite sequence of symbols of the alphabet.	May be empty (length zero)
terminal symbol	A grammar	A symbol that is not to be replaced. It is a symbol of the alphabet of the language being defined.	
transition diagram	An FSA	A diagram defining the next-state function of an FSA	