dont know	know a bit	OK	good!	master	COMS 311 TOPICS		wk6	wk5	wk4	wk3	wk2	wk1
1	2	3	4	5	**BIG-OH**							
					Basics							x
					Definitions of big-oh, omega, theta c>0, n>=0 (7 things)							x
					big-omega							x
					big-theta							x
					Intuition/understanding (graph)							x
					tighter and weaker bounds						x	x
					how to prove O/Omega/Theta techniques							x
					for polynomials choose c >sum of coeff or coeff for omega							x
					for same type compare exponents							x
					take log							x
					Application to Algorithms							x
					ram model (vs actual)							x
					instances and runtime graphs						x	x
					WCET, BCET, ACET						x	x
					Big-oh of code segments						x	x
					problem complexity and algorithmic complexity							x
					code examples of different Os							x
					Big-oh in real world						x	
					real code times (matrix mult)						x	
					effect of cache/pipelining etc						x	
					choosing algo in real-world vs big-Oh						x	
					constants might matter more in real world than O						x	
					easier implementation might make the diff						x	
					Dominance Relationships							x
					logs beat constants							x
					poly beats all logs							x
					exp beats all poly							x
					fact beats all exp							x
					n^n beats fact							x
1	2	3	4	5	**DATA STRUCTURES**							
					Basic						x	x
					arrays (sorted/unsorted)						x	x
					linked lists (singly/doubly; sorted/unsorted)						x	x
					comparison of arrays and linked lists						x	x
					Basic Abstract Data Types						x	x
					Stack, Queue (implementations using array/linkedlists)						x	x
					comparison of operations of diff impl of stack/queue						x	x
					reasons for differences						x	x
					Dictionaries					x	x	
					Hash Tables					x	x	

dont know	know a bit	OK	good!	master	COMS 311 TOPICS		wk6	wk5	wk4	wk3	wk2	wk1
					Other ADTs and their Java Implementations					x	x	
					Binary Search Trees					x	x	
					Priority Search Queues					x	x	
					Heap impl					x	x	
					Fast Heap Impl and Analysis					x		
					Storing points, graphs, sets etc					x	x	
					graphs			x	x	x	x	
					sets				x	x	x	
					big-oh of operations on data structures					x	x	
					algorithms on data structures (BST, HEAP etc)					x	x	
1	2	3	4	5	**GRAPHS (unweighted)**							
					BFS				x	x		
					connected components				x	x		
					two-color problem				x	x		
					DFS on undirected graphs (tree and back edges)			x	x	x		
					articulation vertices (parent, root, bridge cutnodes)			x	x			
					DFS on directed graphs(tree,back,cross,forward)			x	x			
					topological sorting/DAG			x	x			
					strongly connected components			x	x			
					following algorithm by marking graph			x	x			
					big-oh of graph algorithms			x	x	x		
1	2	3	4	5	**Weighted GRAPHS**							
					MST			x				
					prims			x				
					kruskals			x				
					union-find data structure			x				
					reverse delete			x				
					proofs of prims and kruskals			x				
					big-ohs of prims and kruskals			x				
					Shortest Paths			x				
					Dijkstras		x	x				
					proof		x	x				
					big-oh		x	x				
					AllPairs Shortest			x				
					Dijkstras forall nodes		x	x				
					Floyd-Warshalls how algo works + big-oh		x	x				
					Floyd-Warshall's recurrence formula		x	x				
1	2	3	4	5	**P-NP**							
					Intro Concepts							x
					the diagram and four classes of problems				x			x
					informal (solvable, probab intract, provably intract, prov unsolvable)				x			x
					examples of problems in four classes				x			x

dont know	know a bit	OK	good!	master	COMS 311 TOPICS								
								wk6	wk5	wk4	wk3	wk2	wk1
						Halting Problem				x			x
						Hamiltonian Cycle Enumeration problem				x			x
						Hamiltonian Cycle Search problem				x			x
						Independent Set problem				x			x
						Search/Sort problems				x			x
					Classes of problems								
						Undecidable (prove Halting problem is undecidable)				x			x
						P				x			x
						NP				x			x
						prove P is a subset of NP							
						NP-Complete (probably intractable class)							x
						why if a NP-C problem is in P, then P=NP							
						provably intractable classes							x
						NP-Hard problems							
					Reduction								
						optimization, search, decision and reductions			x	x			
						definition of polynomial-time reduction & notation			x	x			
						conclusions (or implications of) from Y <=p X			x	x			
						transitive property (x <=p y, y <=p z => x<=p z)			x	x			
						how to prove X <=p Y (the three steps)			x	x			
						Yes->yes, no-> no, false positives, false negatives etc		x	x				
						interval sched <=p IS constructions and proofs		x	x	x			
						vector cover <=p IS constructions and proofs		x	x	x			
						IS <=p vector cover constructions and proofs		x	x	x			
						bipartite matching <=p IS constructions and proofs		x	x	x			
						given some reduction - ability to prove/disprove yes/no part		x	x				
					NP-Complete								
						Definition and Intuition							
						The first NP-C problem (circuitSAT) & Cook-Levin's theorem							
						how to prove that a problem is NP-C							
						prove 3-SAT is NP-C							
						prove IS is NP-C							
						prove VC is NP-C							
1	2	3	4	5	**PROOF TECHNIQUES**								
					Why is proving important?								x
						Job selection problem							x
						solutions to JS problem							x
						why is proving so important?							x
					CALCULUS								PreRe
						propositional logic rules							PreRe
						predicate logic rules							PreRe
						form of deduction proofs							PreRe

dont know	know a bit	OK	good!	master	COMS 311 TOPICS								
								wk6	wk5	wk4	wk3	wk2	wk1
					DIFFERENT TECHNIQUES							x	reRe
						contradiction (and correct form)			x		x	x	reRe
						induction (form)			x			x	reRe
						direct proof (see form of deduction proofs)			x			x	reRe
						trivial/vacuous						x	reRe
						contrapositive				x		x	reRe
					EXAMPLES IN CLASS								PreRe
						contradiction (and correct form)					x		PreRe
						induction (form)							PreRe
						direct proof (see form of deduction proofs)							PreRe
						trivial/vacuous							PreRe
						contrapositive				x			PreRe
					Proofs in class								
						Halting problem is undecidable					x	x	reRe
						VC <=p IS and IS <=p VC			x	x			
						select jobs satisfies greedy choice + opt substructuring		x					
					Proving Code correct								
						Loop invariants							
						proof of recursive codes..							
1	2	3	4	5	**ALGORITHMIC TECHNIQUES**								
					BRUTE FORCE TECHNIQUES								
						search space for different problems						x	x
						recursion tree for brute force approach						x	x
						back track algo from text book						x	x
						iterative way to generate all subsets							
						recursive way to generate all subsets						x	x
						recursive way to generate all perms						x	x
						recursive way to gen size k subsets						x	x
					DIVIDE AND CONQUER								
						recurrence formula				x	x		
						divide and conquer approach				x	x		
						mergesort + analysis				x	x		
						quicksort + analysis				x	x		
						max				x	x		
						max sum of sequence				x			
						counting inversions				x			
						finding sink in graph							
					Recurrence Formulae					x	x		
						general form of recurrence formula & masters theorem				x	x		
						how to derive recurrence tree, term for each level, sum				x	x		

dont know	know a bit	OK	good!	master	COMS 311 TOPICS			wk6	wk5	wk4	wk3	wk2	wk1
					2T(n/2) + c					x	x		
					T(n/2) + cn					x	x		
					2T(n/2) + cn					x	x		
					3T(n/2) + cn					x	x		
					2T(n/2) + cn^2					x	x		
					5T(n/2) + cn^2					x	x		
					GREEDY TECHNIQUE			x					
					greedy approach			x					
					interval scheduling by greedy approach			x					
					greedy choice property			x					
					optimal substructuring property			x					
					proving gc property			x					
					proving os property			x					
1	2	3	4	5	**DYNAMIC PROGRAMMING**								
					Idea								
					tabular computing (recursive or iterative)			x					
					memoization			x					
					ex: fibonacci			x					
					Finding the solution (i.e. aside from the max/min val)			x					
					Properties								
					diff from Divide & Conquer, Greedy algorithms			x					
					overlapping subproblems			x					
					optimal substructuring property			x					
					examples of greedy-choice property is not satisfied			x					
					Applications (recurrence formula/table/solution)								
					1 min operations			x					
					2 rod-cutting			x					
					3 longest increasing subsequence			x					
					4 min coin change			x					
					5 subset sum			x					
					6 0-1 knapsack			x					
					7 edit-distance								
					8 matrix mult								
					9 weighted interval scheduling (select jobs)								
					10 largest-sum triangle								
					11 Floyd Warshall								
					12 ??								
1	2	3	4	5	**PROGRAMMING ASSIGNMENTS**								
					2 Graduate					x			
					3 Rumors				x				
					4 Larvae Walk			x					
					6 Risk		x						