

# AT: Algorithms Curriculum

Units	Topic breakdown
<b>Unit 0 (~ 2 weeks)</b>	
What is an Algorithm and how do we analyze them?	Can we do better?
Big O notation	loops and nested loops
	How to come up with Recurrence relations & solve them (examples)
<b>Unit 1 (~ 5 weeks)</b>	
Divide and Conquer algorithms (~ 1 week)	Merge Sort with runtime analysis using recurrence equation
Randomized algorithms (~ 2 week)	Quick Sort
Graph Search Algorithms (~ 2 weeks)	BFS, DFS
<b>Unit 2 (~ 5 weeks)</b>	
Dynamic Programming and Memoization	What is Dynamic Programming (DP)? When do we want to use it?
	first example, fibonacci sequence (~ .5 week)
	second example, 0-1 knapsack problem (~.5 week)
	second example, assembly line example (~1 week)
	third example, travelling sales man (~ 1 week)
	lab using dynamic programming?? (~ 2 weeks)
<b>Unit 3 (~ 5 weeks)</b>	
Greedy Algorithms	What is a greedy algorithm? When do we want to use it?
	DP vs greedy
	Review priority queue (~ 1 week)
	Dijkstra's shortest path (~1 week)
	Kruskal's algorithm (~1 week)
	Final lab (~ 2 weeks)