

ICS 211 : Introduction to Computer Science II (3 cr.)			
Description	Algorithms and their complexity, introduction to software engineering, data structures, searching and sorting algorithms, numerical errors.		
Prerequisites	111 , 141 (or concurrent), or consent		
Learning Objectives	<ul style="list-style-type: none"> ● have programmed over 2000 lines of new JAVA code. They will also have recycled some of their classes that may be used by several projects. ● be able to create projects that involve three classes or more. ● understand collection classes and the basics of memory management. ● have written at least one program using Vectors. ● understand the importance of implementing exception handling in their code ● be able to implement the following ADTs and data structures: stacks, queues, linked lists, binary trees, heaps, hash tables ● understand and be able to calculate the Big-O of an algorithm ● have programmed at least one sorting algorithm, and will be familiar with the following sorting algorithms: bubble sort, insertion sort, selection sort, merge sort, heap sort, quick sort ● understand encapsulation, information hiding and ADT concepts and uses ● understand and use and program tree traversal algorithms ● be able to write algebraic equations in prefix, infix and postfix notation using preorder, inorder and postorder traversals ● understand recursion and will be able to write simple recursive methods ● be familiar with Huffman encoding 		
Topic List	#	Topic	Lecture Hours
	1	Procedural abstraction	0.4
	2	ADTs	0.8
	3	Encapsulation and information hiding	0.8
	4	Modularity	1.2
	5	Vectors	2.0
	6	Exception handling	2.0
	7	Big-O analysis	2.4
	8	Introduction to trees	2.4
	9	m-ary trees and their properties	2.4
	10	Applications of trees	1.2
	11	Binary search trees	2.0
	12	Ternary trees	2.4
	13	Recursion	3.2
	14	Sorting algorithms	4.0
	15	Heaps	4.0
	16	Hash tables	4.0
17	Data encoding	4.8	