

Subject Description Form

Subject Code	COMP2011					
Subject Title	Data Structures					
Credit Value	3					
Level	2					
Pre-requisite / Co-requisite / Exclusion	Pre-requisite: COMP1011/COMP1012/ENG2002 Exclusion: COMP2013, DSAI2201					
Objectives	The objectives of this subject are to: <ol style="list-style-type: none"> 1. introduce students to basic concepts of data structures and algorithms; and 2. teach students to apply simple data structures and algorithms in developing computer programs. 					
Intended Learning Outcomes	Upon completion of the subject, students will be able to: <ol style="list-style-type: none"> (a) understand the properties of basic data structures; (b) identify the strengths and weaknesses of different data structures; (c) acquire specialised knowledge of various typical algorithms; (d) design and employ appropriate data structures and algorithms for developing computer applications; and (e) think critically for improvement in the solutions. 					
Subject Synopsis/ Indicative Syllabus	<table border="1" style="width: 100%;"> <tr> <td>Topic</td> </tr> <tr> <td>1. Programming and Algorithms Computer algorithms; types of algorithms; data structures; and abstract data types.</td> </tr> <tr> <td>2. Data Structures: Representation and Algorithms Linear structures: linked-lists, stacks, queues; tree structures: binary trees, balanced trees, tree traversals; and other common data structures: priority queues, heaps.</td> </tr> <tr> <td>3. Sorting Basic sorting algorithms: bubble sort, insertion sort, selection sort; and advanced sorting algorithms: quicksort, mergesort, heapsort.</td> </tr> <tr> <td>4. Searching Common searching algorithms: sequential search, binary search; and advanced searching algorithms: tree search, dictionary and hashing.</td> </tr> </table>	Topic	1. Programming and Algorithms Computer algorithms; types of algorithms; data structures; and abstract data types.	2. Data Structures: Representation and Algorithms Linear structures: linked-lists, stacks, queues; tree structures: binary trees, balanced trees, tree traversals; and other common data structures: priority queues, heaps.	3. Sorting Basic sorting algorithms: bubble sort, insertion sort, selection sort; and advanced sorting algorithms: quicksort, mergesort, heapsort.	4. Searching Common searching algorithms: sequential search, binary search; and advanced searching algorithms: tree search, dictionary and hashing.
Topic						
1. Programming and Algorithms Computer algorithms; types of algorithms; data structures; and abstract data types.						
2. Data Structures: Representation and Algorithms Linear structures: linked-lists, stacks, queues; tree structures: binary trees, balanced trees, tree traversals; and other common data structures: priority queues, heaps.						
3. Sorting Basic sorting algorithms: bubble sort, insertion sort, selection sort; and advanced sorting algorithms: quicksort, mergesort, heapsort.						
4. Searching Common searching algorithms: sequential search, binary search; and advanced searching algorithms: tree search, dictionary and hashing.						

	<p>5. Applications</p> <p>Practical program development using combination of various data structures and algorithms, e.g., friends-book; and efficiency of the various approaches.</p>						
Teaching/ Learning Methodology	<p>The course material will be delivered as a combination of mass lectures and small group supervised tutorial and laboratory sessions. Lectures will provide the required knowledge while tutorials and laboratory sessions allow students to acquire hands-on experience on programming with different algorithms. Programming project provides students with a chance to integrate their knowledge on applying appropriate data structures and algorithms to solve practical problems.</p>						
Assessment Methods in Alignment with Intended Learning Outcomes	Specific assessment methods/tasks	% weighting	Intended subject learning outcomes to be assessed				
			a	b	c	d	e
	Continuous Assessment (assignments / project / quizzes)	30%	✓	✓	✓	✓	✓
	Final Examination	70%	✓	✓	✓	✓	✓
	Total	100%					
Student Study Effort Expected	Class contact:						
	▪ Lecture				39 Hrs.		
	▪ Tutorial/Lab				13 Hrs.		
	Other student study effort:						
	▪ Assignments, Quizzes, Projects, Self-study				55 Hrs.		
Total student study effort				107 Hrs.			
Reading List and References	<p>Reference Books:</p> <ol style="list-style-type: none"> Goodrich, Michael T., Tamassia, Roberto, and Goldwasser, Michael H., <i>Data Structures and Algorithms in Java</i>, 6th Edition, Wiley, 2014. Sedgewick, Robert and Wayne, Kevin, <i>Algorithms</i>, 4th Edition, Addison-Wesley, 2011. Cormen, Thomas H., Leiserson, Charles E., Rivest, Ronald L. and Stein, Clifford, <i>Introduction to Algorithms</i>, 3rd Edition, MIT Press, 2009. 						