

Subject Description Form

Subject Code	COMP2011
Subject Title	Data Structures
Credit Value	3
Level	2
Pre-requisite / Co-requisite / Exclusion	Pre-requisite: COMP1011
Objectives	<p>The objectives of this subject are to:</p> <ul style="list-style-type: none"> • introduce students to basic concepts of data structures and algorithms; and • teach students to apply simple data structures and algorithms in developing computer programs.
Intended Learning Outcomes	<p>Upon completion of the subject, students will be able to:</p> <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) possess the knowledge of various common 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	Topic
	<p>1. Programming and Algorithms</p> <p>Computer algorithms; types of algorithms; data structures; and abstract data types.</p>
	<p>2. Data Structures: Representation and Algorithms</p> <p>Linear structures: linked-lists, stacks, queues; tree structures: binary trees, balanced trees, tree traversals; and other common data structures: priority queues, heaps.</p>
	<p>3. Sorting</p> <p>Basic sorting algorithms: bubble sort, insertion sort, selection sort; and advanced sorting algorithms: quicksort, mergesort, heapsort.</p>
	<p>4. Searching</p> <p>Common searching algorithms: sequential search, binary search; and advanced searching algorithms: tree search, dictionary and hashing.</p>

	<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>																																																										
<p>Teaching/ Learning Methodology</p>	<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>																																																										
<p>Assessment Methods in Alignment with Intended Learning Outcomes</p>	<table border="1" data-bbox="384 555 1479 1205"> <thead> <tr> <th rowspan="2">Specific assessment methods/tasks</th> <th rowspan="2">% weighting</th> <th colspan="5">Intended subject learning outcomes to be assessed (Please tick as appropriate)</th> </tr> <tr> <th>a</th> <th>b</th> <th>c</th> <th>d</th> <th>e</th> </tr> </thead> <tbody> <tr> <td>Continuous Assessment</td> <td>60%</td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td>1. Laboratory Exercises</td> <td>20%</td> <td>✓</td> <td></td> <td>✓</td> <td>✓</td> <td></td> </tr> <tr> <td>2. Programming Project</td> <td>20%</td> <td>✓</td> <td>✓</td> <td>✓</td> <td>✓</td> <td>✓</td> </tr> <tr> <td>3. Test</td> <td>20%</td> <td>✓</td> <td>✓</td> <td>✓</td> <td>✓</td> <td></td> </tr> <tr> <td>Examination</td> <td>40%</td> <td>✓</td> <td>✓</td> <td>✓</td> <td>✓</td> <td>✓</td> </tr> <tr> <td>Total</td> <td>100%</td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table>					Specific assessment methods/tasks	% weighting	Intended subject learning outcomes to be assessed (Please tick as appropriate)					a	b	c	d	e	Continuous Assessment	60%						1. Laboratory Exercises	20%	✓		✓	✓		2. Programming Project	20%	✓	✓	✓	✓	✓	3. Test	20%	✓	✓	✓	✓		Examination	40%	✓	✓	✓	✓	✓	Total	100%					
Specific assessment methods/tasks	% weighting	Intended subject learning outcomes to be assessed (Please tick as appropriate)																																																									
		a	b	c	d	e																																																					
Continuous Assessment	60%																																																										
1. Laboratory Exercises	20%	✓		✓	✓																																																						
2. Programming Project	20%	✓	✓	✓	✓	✓																																																					
3. Test	20%	✓	✓	✓	✓																																																						
Examination	40%	✓	✓	✓	✓	✓																																																					
Total	100%																																																										
<p>Student Study Effort Expected</p>	<p>Class contact:</p> <table border="1" data-bbox="384 1272 1479 1413"> <tr> <td>▪ Lecture</td> <td>39 Hrs.</td> </tr> <tr> <td>▪ Tutorial/Lab</td> <td>13 Hrs.</td> </tr> </table> <p>Other student study effort:</p> <table border="1" data-bbox="384 1480 1479 1621"> <tr> <td>▪ Assignments, Quizzes, Projects, Self-study</td> <td>55 Hrs.</td> </tr> <tr> <td>Total student study effort</td> <td>107 Hrs.</td> </tr> </table>					▪ Lecture	39 Hrs.	▪ Tutorial/Lab	13 Hrs.	▪ Assignments, Quizzes, Projects, Self-study	55 Hrs.	Total student study effort	107 Hrs.																																														
▪ Lecture	39 Hrs.																																																										
▪ Tutorial/Lab	13 Hrs.																																																										
▪ Assignments, Quizzes, Projects, Self-study	55 Hrs.																																																										
Total student study effort	107 Hrs.																																																										
<p>Reading List and References</p>	<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. 																																																										