Title (Units): **COMP2015 Data Structures and Algorithms (3,3,2)**

Course Aims: To develop students' knowledge in data structures and the associated algorithms. To introduce the concepts and techniques of structuring and operating on Abstract Data Types in problem solving. To discuss common sorting, searching and graph algorithms, and to study the complexity and comparisons among these various techniques

Prerequisite: COMP2045 Programming and Problem Solving AND
COMP2046 Problem Solving Using Object Oriented Approach

Course Intended Learning Outcomes (CILOs):
Upon successful completion of this course, students should be able to:

No.	Course Intended Learning Outcomes (CILOs)
	Knowledge
1	Describe the usage of various data structures
2	Explain the operations for maintaining common data structures
3	Recognize the associated algorithms' operations and complexity
	Professional Skill
4	Design and apply appropriate data structures for solving computing problems
5	Develop computer programs to implement different data structures and related algorithms
6	Design simple algorithms for solving computing problems

Calendar Description: This course develops students' knowledge in data structures and the associated algorithms. It introduces the concepts and techniques of structuring and operating on Abstract Data Types in problem solving. Common sorting, searching and graph algorithms will be discussed, and the complexity and comparisons among these various techniques will be studied.

Teaching and Learning Activities (TLAs):

CILOs	Type of TLA
1 - 6	Students will learn the fundamental principles and key concepts via lectures. Tutorials will be conducted to clarify concepts and to have a deeper understanding of the teaching materials, where problems will be given to students for in-depth discussion.
1 - 6	Students will work on written assignments/programming assignments/in-class exercises/test(s)/quizzes to consolidate and apply what they have learnt.
4 - 6	Students will acquire hands-on experience to design and develop data structures and related algorithms via laboratory sections.

Assessment:

No.	Assessment Methods	Weighting	CILOs to be addressed	Description of Assessment Tasks
1	Continuous Assessment	25%	1 - 6	Continuous assessments in the forms of written assignments/programming assignments/in-class exercises/quizzes are designed to measure how well students have learned the basic conception of data structures and algorithms.
2	Practical Test	15%	4 - 6	Practical test(s) are designed to evaluate students' understanding in various data structures and algorithms.
3	Examination	60%	1 - 6	Final examination questions are designed to see how far students have achieved their intended learning outcomes. Questions will primarily be analysis and skills based to assess students' ability in understanding and application of various data structures and algorithms.

Assessment Rubrics:

Level of Achievement	Elaboration on Course Grading Description
Excellent (A)	The student's performance is outstanding in almost all the intended course learning outcomes.
Good (B)	The student's performance is good in most of the intended course learning outcomes.
Satisfactory (C)	The student's performance is satisfactory. It largely meets the intended course learning outcomes.
Marginal Pass (D)	The student's performance is barely satisfactory. It marginally meets the intended course learning outcomes.
Fail (F)	The student's performance is inadequate. It fails to meet any of the intended course learning outcomes.

Course Content and CILOs Mapping:

Content		CILO No.
I	Algorithm Analysis	3
II	Abstract Data Types	1,2,4,5
III	Trees	1,2,4,5
IV	Hashing	2-6
V	Priority Queues (Heaps)	2-6
VI	Sorting	3, 5-6
VII	Graph Algorithms	3, 5-6

References:

- M.A. Weiss, Data Structures and Algorithm Analysis in Java, 3rd Edition, Addison-Wesley, 2011.
- M.A. Weiss, Data Structures and Algorithm Analysis in C++, 4th Edition, Addison-Wesley, 2013.
- Thomas H. Cormen, Introduction to Algorithms, 3rd Edition, MIT Press, 2009.

Course Content:

Topic

I. Algorithm Analysis
   A. Mathematical background
   B. Recursions
   C. Big-o notations
   D. Running time calculation

II. Abstract Data Types
   A. Lists
   B. Stacks
   C. Queues

III. Trees
   A. Tree traversals
   B. Binary trees and binary search trees
   C. AVL trees
   D. B-trees

IV. Hashing

A. Hash function
B. Separate chaining
C. Open addressing
D. Rehashing

V. Priority Queues (Heaps)
A. Binary heaps
B. Applications of priority queues
C. D-heaps

VI. Sorting
A. Insertion sort
B. Shell sort
C. Heap sort
D. Merge sort
E. Quick sort
F. Bucket sort
G. External sorting

VII. Graph Algorithms
A. Topological sort
B. Shortest-path algorithms
C. Minimum-span tree