DRAFT

Data Structures and Algorithms

Credit Hours: 3-4 Semester Hours
Prerequisite: OSC002 Computer Science II or an equivalent object oriented programming course
Related TAG: Computer Science
Student Learning Outcomes marked with an asterisk (*) are considered essential and must be covered:
Learning outcome 1. Demonstrate proficiency in simple numerical algorithms such as computing the average of a list of numbers, finding the min, max, and mode in a list, calculating factorial, approximating the square root of a number, or finding the greatest common divisor.
Learning outcome 2. Demonstrate the use of basic data structures and explain their design and use case scenarios including arrays, linked lists, stacks, queues, priority queues, graphs, and trees.*
Learning outcome 3. Implement hash tables including strategies for avoiding and resolving collisions.*
Learning outcomes 4. Calculate the time and space efficiencies of an algorithm and underlying data structures using asymptotic analysis such as big-O notation, asymptotic notation, average-case, amortized, and worst-case analysis.*
Learning outcomes 5. Implement and compare the complexities of various sorting and searching algorithms such as bubble sort, insertion sort, merge sort, quicksort, binary search, and linear search.*
Learning outcomes 6. Use and understand time and space complexities of tree and graph traversal and shortest path algorithms such as depth-first search, breadth-first search, Dijkstra's, and Bellman-Ford.*
Learning outcome 7. Implement advanced data structures such as balanced binary trees, heaps, and tries.