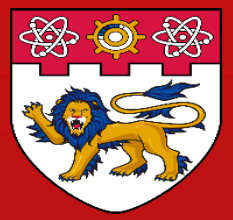


NANYANG TECHNOLOGICAL UNIVERSITY
SINGAPORE

SC1007: DATA STRUCTURES AND ALGORITHMS

Course Introduction

College of Engineering
School of Computer Science and Engineering

INSTRUCTOR INFORMATION

- Dr Newton FERNANDO (ofernando@ntu.edu.sg)
  - Course coordinator
  - Research background: Human-computer interaction, multimodal and natural interfaces, crowdsourcing
  - Room No: N4-2C-80

- Dr Liu Siyuan (SYLiu@ntu.edu.sg)
  - Research background: Explainable AI, Trust and reputation management, Serious games
  - Room No: N4-02c-117a

COURSE SCHEDULE (LECTURES, LABS, TUTORIALS AND ASSIGNMENTS)

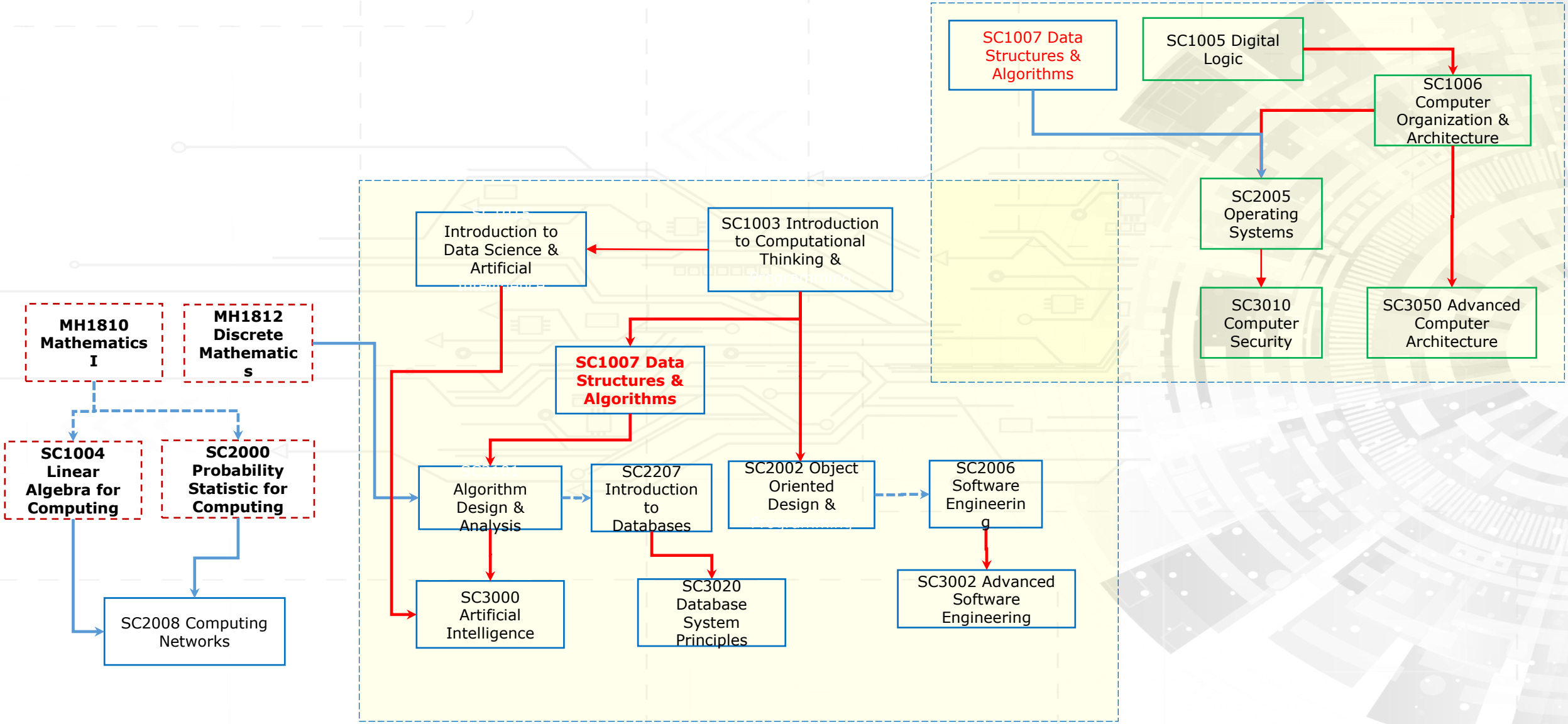
Week	Topic	Tutorials	Labs	Assignment Deadlines
1	Introduction to Data Structure & Algorithm			
2	Introduction Linked List (LL)			
3	Linked List (LL) – Linear Search		Lab 1 (LL)	
4	Stack and Queue (SQ) – Arithmetic Operations	T1 (LL)	Lab 2 (SQ)	AS1 (LL) **(09/02/2024)**
5	Binary Trees (BT) and Binary Search Trees			
6	Binary Trees - Binary Search and AVL Trees	T2 (SQ)	Lab 3 (BT)	AS2 (SQ) **(23/02/2024)**
7	Analysis of Algorithm (AA)	T3 (BT & BST)	Lab 4 (BST)	AS3 (BT and BST) **(01/03/2024)**
	Lab Test 1 (Recess Week: 05/03/2024 - 06/03/2024)			
8	Hash Table			
9	Basic Graph (G)	T4 (AA)	Lab 5 (Hash Table)	
10	DFS + backtracking/ Permutation	T5 (Hash Table)	Lab 6 (Graph)	AS 4 (Hash Table) **(29/03/2024)**
11	Dynamic Programming		Lab 7 (Backtracking)	
12	Permutation/Matching	T6 (Graph)	Lab 8 (Dynamic Programming)	AS 5 (Graph) **(12/04/2024)**
13	Revision			AS 6 (Permutation/ Matching) **(19/04/2024)**
14	**Lab Test 2 + Quiz (23/04/2024 - 24/04/2024)**			

COURSE SCHEDULE (PART-TIME)

Week	Date	1830-1930	1930-2030	2030-2130	Venue
1	18-Jan-24	Introduction To Data Structure and Algorithm	Introduction Linked List (LL)	Introduction Linked List (LL)	**TR +3**
2	25-Jan-24	Linked List (LL) – Linear Search		T1 (LL)	**TR +3**
3	01-Feb-24	Stack and Queue (SQ) - Arithmetic Expression	Lab 1 (LL)	Lab 1 (LL)	Software Lab 2
4	08-Feb-24	Stack and Queue (SQ) - Arithmetic Expression	Lab 2 (SQ)	Lab 2 (SQ)	Software Lab 2
5	15-Feb-24	Tree Traversal - Binary Search	Tree Traversal - Binary Search	T2 (SQ)	**TR +3**
6	22-Feb-24	Binary Trees - Binary Search and AVL Trees	Lab 3 (BT)	Lab 3 (BT)	Software Lab 2
7	29-Feb-24	T3 (BT & BST)	Lab 4 (BST)	Lab 4 (BST)	Software Lab 2
Recess	07-Mar-24	Lab Test 1			Software Lab 2
8	14-Mar-24	Hash Table + Graph Representation		BFS, DFS	**TR +3**
9	21-Mar-24	Backtracking, Permutation	Lab 5 (Hash Table)	Lab 5 (Hash Table)	Software Lab 2
10	28-Mar-24	T4 (AA)	Lab 6 (Graph)	Lab 6 (Graph)	Software Lab 2
11	04-Apr-24	Dynamic Programming	Lab 7 (Backtracking)	Lab 7 (Backtracking)	Software Lab 2
12	11-Apr-24	T5 (Hash Table)	Lab 8 (Dynamic Programming)	Lab 8 (Dynamic Programming)	Software Lab 2
13	18-Apr-24	Permutation/Matching	Permutation/Matching	T6 (Graph)	**TR +3**
14	25-Apr-24	Lab Test 2 and Final Quiz			Software Lab 2

Assessments	Weighting
Assignments (6 Assignments)	40%
Lab Tests (Two Lab Tests):	40%
Quiz (1 Quiz)	20%

CS PROGRAMME STRUCTURE



LEARNING OBJECTIVES

- Lectures focus on introduction to concepts
- Tutorials focus on understanding the concepts
- Lab Sessions focus on practice and realization
- Assignments and Lab Tests are assessments

LEARNING OUTCOMES

- Select appropriate data structures and Algorithm
- Implement algorithms to solver real world problems using C Language
- Conduct complexity analysis of algorithms

OVERVIEW OF SC1007

- **Data Structures**:
  - Concepts of pointers and structures (aggregates)
  - Introduce some classical data structures
    - Linear: Linked list, stack, queue
    - Nonlinear: tree
  - Implement these data structures
- **Algorithms**:
  - Analysis of Algorithm time complexity and space complexity
  - Introduce to some typical algorithms and their applications
  - Introduce to some algorithm design strategies
- **Implementation**:
  - C programming

GOALS

"I will, in fact, claim that the difference between a bad programmer and a good one is whether he considers his code or his data structures more important. Bad programmers worry about the code. Good programmers worry about data structures and their relationships."

Linus Torvalds, 2006

(Creator of the Linux kernel)