NUS School of Computing
Master of Computing (General Track) - Essential and Capstone Preparation Modules
Semester 2, AY2021/2022

Module Code & Title	Lecture Day/Period	Time	Venue	Remarks	Exam
IT5001 Software Development Fundamentals *Lecturer: Alan Cheng and Jenny Lim*	Wednesday [12 Jan 2022 - 2 Mar 2022]	4pm - 6pm	**LECTURE** (E-Learn)	All students are to attend both Wed 4pm-6pm and Fri 2pm-3pm lessons. Part-time students may attend the Wed 6.30pm-8.30pm and Sat 9am-10am lessons instead.	Saturday, 19 Mar 2022, 2pm - 4pm For exam details, please check with your lecturer and LumiNUS for announcements
	Friday [14 Jan 2022 - 4 Mar 2022]	2pm - 3pm	**LECTURE** (E-Learn)		
	Monday [17 Jan 2022 - 7 Mar 2022]	2pm - 4pm	**LAB** (E-Learn)	Students attending the Wed 4pm-6pm and Fri 2pm-3pm lessons will attend either lab session.	
		4pm - 6pm	**LAB** (E-Learn)		
	Wednesday [12 Jan 2022 - 2 Mar 2022]	6.30pm - 8.30pm	**LECTURE** (E-Learn)	Part-time students may attend both Wed 6.30pm-8.30pm and Sat 9am-10am lessons.	
	Saturday [15 Jan 2022 - 5 Mar 2022]	9am - 10am	**LECTURE** (E-Learn)		
	Saturday [15 Jan 2022 - 5 Mar 2022]	10am - 12pm	**LAB** (E-Learn)	Part-time students attending Wed 6.30pm-8.30pm and Sat 9am-10am lessons will attend this lab session.	
IT5003 Data Structures and Algorithms *Lecturer: Jenny Lim and Steven Halim*	Wednesday [9 Mar 2022 - 27 Apr 2022]	6.30pm - 8.30pm	**LECTURE** (E-Learn)	Students are to attend both Wed and Sat lectures.	Saturday, 7 May 2022, 9am - 11am For exam details, please check with your lecturer and LumiNUS for announcements
	Saturday [12 Mar 2022 - 30 Apr 2022]	9am - 10am	**LECTURE** (E-Learn)		
	Saturday [12 Mar 2022 - 30 Apr 2022]	10am - 12pm	**LAB** (E-Learn)	Students may select one lab session to attend.	
	Monday [14 Mar 2022 - 2 May 2022]	2pm - 4pm	**LAB** (E-Learn)		

		4pm - 6pm	**LAB** (E-Learn)		
IT5004 Enterprise Systems Architecture Fundamentals Lecturer: Lek Hsiang Hui	Thursday [13 Jan 2022 - 14 Apr 2022]	6.30pm - 9.30pm	**LECTURE** (E-Learn)	Students are to attend all lectures.	Thursday, 5 May 2022, 9am - 11am For exam details, please check with your lecturer and LumiNUS for announcements
IT5006 Fundamentals of Data Analytics Lecturer: Ashish Deepak Dandekar	Friday [14 Jan 2022 - 15 Apr 2022]	6.30pm - 9.30pm	**LECTURE** (E-Learn)	Students are to attend all lectures.	Friday, 6 May 2022, 2.30pm - 4.30pm For exam details, please check with your lecturer and LumiNUS for announcements
IT5007 Software Engineering on Application Architecture Lecturer: Prasanna Karthik Vairam	Tuesday [11 Jan 2022 - 12 Apr 2022]	4pm - 7pm	**LECTURE** (E-Learn)	Students are to attend all lectures.	No Exam
IT5100A Typed Functional Programming in Practice Lecturer: Chin Wei Ngan	Wednesday [12 Jan 2022 - 23 Feb 2022]	9am - 11am	**LECTURE** (E-Learn)	Students are to attend all lectures.	No Exam
IT5100B High-Throughput Stream Programming Lecturer: Razvan Voicu	Wednesday [2 Mar 2022 - 13 Apr 2022]	9am - 11am	**LECTURE** (E-Learn)	Students are to attend all lectures.	No Exam

* Modules offered, descriptions and schedules may be subject to change.
*Please refer to this link for lecture/lab venues: https://www.comp.nus.edu.sg/maps/venues/

Essential Modules:

IT5001 Software Development Fundamentals
This module aims to introduce non-computing students to the principles and concepts of software development at an accelerated pace. Students will be introduced to the basics of programming (control flow, code and data abstraction, recursion, types, OO), development methodology (ensuring correctness, testing, debugging), simple data structures and algorithms (lists, maps, sorting), and software engineering principles. Through hands on assignments and projects, students will learn good software development practices (documentation, style) and experience a typical software engineering cycle (waterfall and agile workflow).

IT5002 Computer Systems and Applications
This module aims to introduce non-computing students to (a) the common principles and concepts in computer systems: abstraction, layering, indirection, caching, hierarchical naming, prefetching, pipelining, locking, concurrency; (b) the inner workings of a computing device, including hardware (CPU, memory, disks), operating systems (kernels, processes and threads, virtual memory, files), and applications (Web, databases).

IT5003 Data Structures and Algorithms

This module introduces non-computing students to efficient computational problem solving in an accelerated pace. Students will learn to formulate a computational problem, identify the data required and come up with appropriate data structures to represent them, and apply known strategies to design an algorithm to solve the problem. Students will also learn to quantify the space and time complexity of an algorithm, prove the correctness of an algorithm, and the limits of computation. Topics include common data structures and their algorithms (lists, hash tables, heap, trees, graphs), algorithmic problem solving paradigms (greedy, divide and conquer, dynamic programming), and NP-completeness.

IT5004 Enterprise Systems Architecture Fundamentals

This module aims to equip non-computing students with fundamental knowledge in architecting and designing modern Enterprise Systems in organisations that can be reasonably complex, scalable, distributed, component-based and mission-critical. Students will develop an understanding of high-level concepts such as enterprise architecture and software architecture. They will them move on to acquire fundamental systems analysis and design techniques such as object-oriented requirements analysis and design using the Unified Modelling Language.

IT5005 Artificial Intelligence

The study of artificial intelligence, or AI, aims to make machines achieve human-level intelligence. This module provides a comprehensive introduction to the fundamental components of AI, including how problem-solving, knowledge representation and reasoning, planning and decision making, and learning. The module prepares students without any AI background to pursue advanced modules in AI.

IT5006 Fundamentals of Data Analytics

This module introduces students to the fundamental concepts in business analytics. They can learn how to apply basic business analytics tools (such as R), and how to effectively use and interpret analytic models and results for making informed business decisions. The module prepares students without any analytics background to pursue advanced modules in business and data analytics.

IT5007 Software Engineering on Application Architecture

Pre-requisite: IT5003 Data Structures and Algorithms

To meet changing business needs, this course focuses on flexible and agile software development on modern application architecture. Students learn to design and develop modern applications that support multiple clients across different platforms such as desktop, mobile devices and cloud. The course covers designing (1) website-based front-end software and (2) mobile app front-end that interacts with a common cloud-based backend. The final part involves engineering software for higher-level objectives such as security and performance. Tools and techniques for writing modern software, such as, HTML5, CSS3, React.js, Node.js, MySQL/MongoDB, and Git will be taught.

Capstone Preparation Modules:

IT5100A Typed Functional Programming in Practice

Typed functional programming are becoming more widely adopted in industry, as can be seen in the success of a number of advanced programming languages, such as OCaml, Haskell and Scala 3. These advanced languages offer a range of expressive features to allow robust, reusable and high-performing software codes to be safely and rapidly developed. This module will cover key programming techniques of typed functional programming that are becoming widely adopted, such as strong typing, code composition and abstraction, effect handlers, and safe techniques for asynchronous and concurrent programming.

IT5100B High-Throughput Stream Programming

The global availability of data has reached a level where aggregating data into generic, general-purpose "stores" is no longer feasible. Having data collections statically available for querying by interested parties on demand is increasingly becoming the way of the past. Instead, a new paradigm, called Data Streaming, has emerged recently. In this paradigm, data is bundled into high-throughput "streams'' that are sharded efficiently across a large number of network nodes. Consumers, sometimes counted in hundreds of thousands, or millions, "subscribe" to data subsets and are notified when new data becomes available, being under the obligation to process it immediately, or lose it. Consequently, data storage is no longer centralized, but rather distributed into many smaller-sized abstract collections. This new approach to "Big Data" requires a new set of tools, platforms, and solution patterns. In this course we propose to explore several facets of this new paradigm: • The Stream paradigm introduced in Java 8. • Platforms that implement Data Streaming, such as Kafka, and the Java bindings in the library KafkaConnect. • Computing paradigms for stream processing, such as Reactive Programming, and the library RxJava. • High-performance stream computing platforms, such as Flink. The course will be using Java as the main vehicle for introducing concepts and showcasing examples.