Annexe A: New/Revised Course Content in OBTL+ Format

Course Overview

Expected Implementation in Academic Year	2025-26
Semester/Trimester/Others (specify approx. Start/End date)	Semester 2
Course Author * Faculty proposing/revising the course	Lee-Chua Lee Hong
Course Author Email	clhlee@ntu.edu.sg
Course Title	Introduction to Computational Thinking
Course Code	CV1014
Academic Units	3
Contact Hours	39
Research Experience Components	Not Applicable

Course Requisites (if applicable)

Pre-requisites	
Co-requisites	
Pre-requisite to	
Mutually exclusive to	
Replacement course to	
Remarks (if any)	

Course Aims

Computational thinking (CT) is a problem solving process with the aid of computer; i.e. formulating a problem and expressing its solution in such a way that a computer can effectively carry it out. It includes a number of characteristics, such as breaking a problem into small and repetitive ordered steps, logically ordering and analyzing data and creating solutions that can be effectively implemented as algorithms running on computer. As such, computational thinking is essential not only to the Computer Science discipline, it can also be used to support problem solving across all disciplines, including math, science, engineering, business, finance and humanities.

The aim of this course is hence to take students with no prior experience of thinking in a computational manner to a point where you can derive simple algorithms and code the programs to solve some basic problems in your domain of studies. In addition, the course will include topics to appreciate the internal operations of a processor, and raise awareness of the socio-ethical issues arising from the pervasiveness of computing technology.

Course's Intended Learning Outcomes (ILOs)

Upon the successful completion of this course, you (student) would be able to:

ILO 1	Describe the internal operation of a basic processor, how a program is executed by a computer and computing trends.
ILO 2	Code basic programs based on the programming language used in the course
ILO 3	Formulate a problem and express its solution in such a way that a computer can effectively carry it out. (i.e. equip you with CT skills)
ILO 4	Apply the CT concepts on case studies/problem-based scenarios through hands-on practice of the CT processes.

Course Content

No	Topics
1	**Course Overview and Concepts of Computational Thinking** Solving complex problem using computer **-** enables the student to work out exactly what to tell the computer to do.
2	**Overview of Programming Languages** Graphic programming, high level programming languages (Python, C, Java, R, Matlab)

3	**Basic internal operation of computer** Basic computer organization (Processor, Memory, I/O) and how a computer execute a program (Machine instructions)	Reading and
4	**Basic program structure: control constructs and data types** Concepts of data types, variables; Pseude code and flowcharts; Sequences, Selection (if/else), iteration (for/while loop);	
5	**CT concept - Abstraction** Problem formulation - reducing something to a very simple set of characteristics to only focusing on the most relevant to the problem. Concept of functions/libraries and data structure	
6	**CT concept - Decomposition** Break a complex problem into smaller and more manageable parts/steps, such that each of these smaller problems can then be looked at individually	
7	**CT concept – Pattern recognition** Looking for similarities among and within problems, which also enable re-use knowledge of previous similar problems	
8	**CT concept – Algorithm** Reformulating the problem into series of ordered steps through Identifying, analyzing, and implementing possible solutions with the goal of achieving the most efficient and effective combination of steps and resources. (Some common/useful examples: Sorting and searching)	
9	**Limit of computing** Analysis of Algorithm Complexity to determine how much resources (space and time) are needed to execute an Algorithm in order to achieve code optimization.	
10	**Computing Trends** Cloud, Edge and Fog computing, Quantum Computers	

11	**Social-Ethical Issues and Ramifications of Computing** Fairness, Privacy, Sharing, Hacking, software Piracy, Data Protection, Cyberbullying and trolling, Fake news, digital divides, IP/Copyright

References (if applicable)

The course will not use any specific text book. The following books and websites will be used as reference materials.

1. The Practice of Computing using Python; Enbody, Richard, and William Punch, Pearson, 2021.
2. Introduction to Computation and Programming Using Python : With Application to Understanding Data; (3rd Ed) John V. Guttag, MIT Press Ltd, 2021.
3. https://edu.google.com/resources/programs/exploring-computational-thinking/

The teaching faculty will furnish the updated reference if there are newer versions.

Planned Schedule

Week or Session	Topics or Themes	ILO	Readings	Delivery Mode	Activities
1	Computational Thinking Concepts, Programming languages	3, 4		In-person	On-line Video, Programming Languages exercises
2	Basic internal operation of computer	1		In-person	On-line Video, Computer operations exercises
3	Basic program structure: Case Study, Pseudo code and flowchart	2		In-person	On-line Video, Python programming exercises
4	Basic program structure: Data type, Variable, sequence, logic and comparison operation	2		In-person	On-line Video, Python programming exercises
5	Basic program structure: Selection and Iteration	2		In-person	On-line Video, Python programming exercises
6	Procedural abstraction: function and library	2, 3		In-person	On-line Video, Python programming exercises (Function)
7	Data abstraction: Data structure	2, 3		In-person	On-line Video, Python programming exercises (Function/Data Structures)

Week or Session	Topics or Themes	ILO	Readings	Delivery Mode	Activities
8	Decomposition Case study	2, 3, 4		In-person	On-line Video, Mini project – Flow Chart Design
9	Pattern recognition Case study	2, 3, 4		In-person	On-line Video, Mini project – Coding
10	Algorithms, Sorting algorithm	2, 3, 4		In-person	On-line Video, Mini project – Coding and debugging
11	Algorithm design, Searching algorithm	2, 3, 4		In-person	On-line Video, Mini project – Coding and debugging
12	Algorithm Complexity Analysis Big-0 concept	2, 3, 4		In-person	On-line Video, Mini project – Testing and assessment
13	Computing trends and Ethical considerations	1		In-person	On-line Video, Mini project – assessment

Learning and Teaching Approach

Approach	How does this approach support you in achieving the learning outcomes?
TEL (Online Video)	Topics will be delivered as a series of online videos lectures, and you will also be provided reference reading materials for self-study to achieve LO 1 to LO 4.
Example Class (Face-to-face)	The Example class will be used as seminar sessions for students to clarify the contents of the online topic, as well as hands-on sessions to equip you with practical knowledge on coding, and on the design and implementation of a mini project to achieve LO 1 to LO 4.

Assessment Structure

Assessment Components (includes both continuous and summative assessment)

No.	Component	ILO	Related PLO or Accreditation	Weightage	Description of Assessment Component	Team/Individual	Rubrics	Level of Understanding
1	Continuous Assessment (CA): Others(CA1: Online participation)	2,3,4	a, b, c	20	Students will complete this section through online at their own time	Individual	Analytic	Multistructural
2	Continuous Assessment (CA): Others(CA2: Hands-on exercise)	1, 2, 3, 4	a, b, c, d	10	exercise conducted in the lab	Individual	Analytic	Multistructural
3	Continuous Assessment (CA): Test/Quiz(CA3: Quiz 1)	2,3,4	a, b, c	20	Quiz session conducted in the lab	Individual	Analytic	Multistructural
4	Continuous Assessment (CA): Test/Quiz(CA4: Quiz 2)	2,3,4	a, b, c	20	Quiz session conducted in the lab	Individual	Analytic	Multistructural

No.	Component	ILO	Related PLO or Accreditation	Weightage	Description of Assessment Component	Team/Individual	Rubrics	Level of Understanding
5	Continuous Assessment (CA): Project(CA5: Mini Project)	2, 3, 4	a, b, c, d, e, i, j	30	Students form group of 3 to demonstrate 1 working program in the form of a mini project In order to account for Individual Contribution to the project, the Modification Factor (MF) will be applied, and the MF is derived from peer assessment, more details could be found in the rubrics file	Team	Holistic	Relational

Description of Assessment Components (if applicable)

Formative Feedback

For online tasks, immediately after you submitted the answers, you will see your scores, your answers, the correct answers, feedback on your incorrect answers, and explanations for the correct answers. For online and MCQ quizzes, individual feedback will be provided to students through evaluation of their submissions. Quiz answers will be discussed in the example class. you will also see the average scores of the other students in the same cohort.

For lab assessments, you will be given verbal feedbacks during your demonstrations of the circuits.

NTU Graduate Attributes/Competency Mapping

This course intends to develop the following graduate attributes and competencies (maximum 5 most relevant)

Attributes/Competency	Level
Collaboration	Intermediate
Creative Thinking	Intermediate
Problem Solving	Advanced
Transdisciplinarity	Advanced

Course Policy

Policy (Academic Integrity)

Good academic work depends on honesty and ethical behaviour. The quality of your work as a student relies on adhering to the principles of academic integrity and to the NTU Honour Code, a set of values shared by the whole university community. Truth, Trust and Justice are at the core of NTU's shared values. As a student, it is important that you recognize your responsibilities in understanding and applying the principles of academic integrity in all the work you do at NTU. Not knowing what is involved in maintaining academic integrity does not excuse academic dishonesty. You need to actively equip yourself with strategies to avoid all forms of academic dishonesty, including plagiarism, academic fraud, collusion and cheating. If you are uncertain of the definitions of any of these terms, you should go to the academic integrity website for more information. On the use of technological tools (such as Generative AI tools), different courses / assignments have different intended learning outcomes. Students should refer to the specific assignment instructions on their use and requirements and/or consult your instructors on how you can use these tools to help your learning. Consult your instructor(s) if you need any clarification about the requirements of academic integrity in the course.

Policy (General)

As a student of the course, you are required to abide by both the University Code of Conduct and the Student Code of Conduct. The Codes provide information on the responsibilities of all NTU students, as well as examples of misconduct and details about how students can report suspected misconduct. The university also has the Student Mental Health Policy. The Policy states the University's commitment to providing a supportive environment for the holistic development of students, including the improvement of mental health and wellbeing. These policies and codes concerning students can be found in the following link.
http://www.ntu.edu.sg/SAO/Pages/Policies-concerning-students.aspx

Policy (Absenteeism)

0

Policy (Others, if applicable)

0

Last Updated Date: 02-10-2025 00:15:16

Last Updated By: YANG En-Hua

Rubrics for CV1014 Introduction to Computational Thinking

1. Assessment Criteria for Mini Project (CA5)

You will submit the code(s) for data analysis, the visualization dashboard, and make a presentation to illustrate the Mini Project. Mini-Project will be graded out of 100 points, which will then be scaled down to 30% of your total marks.

Criteria	**Standards**		
	Fail standard (0-39%)	**Pass standard (40-80 %)**	**High standard (81-100 %)**
Demonstrate (including explanation) the use of CT concepts in the implementation of the project. (LO 1,2,3,4)	Demonstrated less than 40% of the functionalities according to the specifications.	Demonstrated 40% to 80% of the functionalities according to the specifications.	Demonstrated more than 80% of the functionalities according to the specifications.
Individual assessment (LO 1,2,3,4)	Unable to explain individual work performed. Unable to understand and answer questions.	Reasonable depth of explanation on individual work performed. Able to understand and answer some questions.	Offer good explanation on individual work performed. Able to understand and answer most questions.

2. Peer Evaluation of Contributions to the Mini Project

	Performance Level/Criteria			
Performance Indicators	**Outstanding: 4**	**Good: 3**	**Average, meet expectation: 2**	**Below expectations: 1**
Collaborative behaviour	Cooperative and always delivered assigned tasks on time. Take initiative to help other to ensure success of team project.	Cooperative and always delivered assigned tasks on time. Willing to assist others upon request.	Stop short at delivering assigned tasks, sometimes after reminder(s).	Uncooperative, non-committed, always miss deadlines.
Quality of works	Quality of works higher than overall group quality, or go extra miles to assist teammate to enhance the quality of group works.	Good quality of deliverables under individual responsibility.	Acceptable quality of deliverables under individual responsibility.	Quality of works not acceptable.

Ideas & participations	Active participation and initiatives, good ideas & suggestions in enhancing the quality of group works.	Contributed suggestions and ideas to enhance the quality of group works.	Somewhat contributed in enhancing the quality of group works.	Did not participate in group works.

Average Peer Assessment Score	**MF**
3.51 to 4.00	1.05
2.76 to 3.50	1.00
2.51 to 2.75	0.95
2.00 – 2.50	0.9
Below 2.0	Separate Assessment

Peer assessment exercise will be anonymous and done towards the end of the semester.

For student who has average peer assessment score below 2.0, the coordinator might contact them, and/or contact any other group member(s) to further assess the appropriate MF.