Subject Description Form

Subject Code	COMP2421
Subject Title	Computer Organization
Credit Value	3
Level	2
Pre-requisite / Co-requisite / Exclusion	
Objectives	The objective of this subject is to provide students with an introductory but comprehensive knowledge on computer systems, computer organisation, computer system architecture and assembly language programming.
Intended Learning Outcomes	Upon completion of the subject, students will be able to: *Professional/academic knowledge and skills* (a) understand the organisation of a modern computer system and be able to relate them to real examples implemented in commercially successful products; (b) understand the internal organisation of a computer system through practicing with an assembly language; (c) apply concepts and skills to solve real life problems using a low-level programming language; *Attributes for all-roundedness* (d) provide framework for thinking about computer organisation; and (e) be aware of the advancement of computing systems development.
Subject Synopsis/ Indicative Syllabus	

Topic
1. Basic Concepts Introduce basic concepts of computer organisation and architecture, basic components and functions, clock and synchronisation, instruction cycles, etc.
2. Computer Arithmetic Binary system and arithmetic; signed integers and representation; octal and hexadecimal systems; BCD representation; conversion between representations; floating point representations.
3. CPU and Assembly Language Instruction sets, characteristics and functions; CPU structure and functions; reduced instruction set computers; assembler commands; program instructions; assembler and execution of programs; assembly language programming.

4. Boolean Algebra and Logic Networks

Boolean algebra and networks; basic logical operations; derivation of logical expressions; logic gates; flip-flops; counters; half and full adders.

5. Memory, I/O and Storage Devices

Input and output devices; interconnecting system components; interfacing; buses; interrupts in I/O systems; standard bus interfaces; main memory; RAM; ROM; secondary storage; cache memory; virtual memory and operating systems support.

Teaching/ Learning Methodology

Lectures teach students on the main concepts of the course, together with comprehensive examples, and class questions and answers for easy understanding.

Tutorials and lab sessions offer the opportunity for students to review the lecture materials through online exercises and also the use of programming tools to learn to program.

Programming assignments will give students the opportunity to solve problems through implementation where they understand and practice on how programs can be written and compiled to run to complete certain tasks.

Homework assignments help students to develop analytical and problem solving skills.

Assessment Methods in Alignment with Intended Learning Outcomes

Specific assessment methods/tasks	% weighting	Intended subject learning outcomes to be assessed				
		a	b	c	d	e
Continuous Assessment	**55%**					
1. Homework Assignments		✓			✓	
2. Online QA Exercises		✓			✓	
3. Programming Assignments		✓	✓	✓	✓	✓
4. Quizzes and Mid-Terms		✓	✓	✓	✓	
Examination	**45%**	✓	✓	✓	✓	
Total	100%					

Explanation of the appropriateness of the assessment methods in assessing the intended learning outcomes:

All five items are relevant to the assessment of (a) understand the organisation of a modern computer system and be able to relate them to real examples implemented in commercially successful products and (d) provide framework for thinking about computer organisation.

	Programming exercises in assignments are used to assess programming skills in (b) understand the internal organisation of a computer system through practicing with an assembly language; and (c) apply concepts and skills to solve real life problems using a low level programming language. The programming skills learnt can also help student in (e) continue the lifetime learning necessary for staying at the forefront of computing systems development. The quizzes and the examination are also used to assess the programming skills learnt (for items b, c).	
Student Study Effort Expected	Class contact:	
	▪ Lecture	39 Hrs.
	▪ Laboratory	13 Hrs.
	Other student study effort:	
	▪ Reading to Understand the Concepts	40 Hrs.
	▪ Homework and Programming Assignments, Online QA, and Preparation for Quizzes and Final Exam	28 Hrs.
	Total student study effort	120 Hrs.
Reading List and References	**Textbook:** 1. Stallings, W., *Computer Organization and Architecture: Designing for Performance*, 10th Edition, Pearson, 2016. **Reference Books:** 1. Patterson, David A. and Hennessy, John L., *Computer Organization and Design: The Hardware/Software Interface*, 5th Edition, Morgan Kaufmann, 2014. 2. Mano, M.M. and Kime, C.R., *Logic and Computer Design Fundamentals*, 5th Edition, Pearson, 2016. 3. Hamacher, C., Vranesic, Z. and Zaky, S., *Computer Organization*, 6th Edition, McGraw-Hill, 2011. 4. Brey, Barry B., *The Intel Microprocessors: 8086/8088, 80186/80188, 80286, 80386, 80486, Pentium, Pentium Pro, and Pentium II Processors: Architecture, Programming, and Interfacing*, 8th Edition, Prentice Hall, 2009. 5. Antonakos, J.L., *The 68000 Microprocessor*, 5th Edition, Prentice Hall, 2003. 6. Kane, Gerry and Heinrich, Joe, *MIPS RISC Architecture*, 2nd Edition, Prentice Hall, 1998.	