Subject Description Form

Subject Code	COMP201
Subject Title	Principles of Programming
Credit Value	3
Level	2
Pre-requisite/ Co-requisite/ Exclusion	Pre-requisite/Co-requisite: Nil Exclusion: ENG236
Objectives	• To provide students with knowledge on the fundamental elements in computer programming. • To introduce advanced computer programming techniques necessary for developing more sophisticated computer application programs.
Intended Learning Outcomes	Upon completion of the subject, students will be able to: *Professional/academic knowledge and skills* (a) understand the programming elements for solving computing related problems; (b) possess the ability to design and develop efficient computer programs for solving problems; (c) possess the ability to learn advanced programming techniques independently; (d) possess the ability to learn other high level programming languages independently; *Attributes for all-roundedness* (e) develop skills in problem solving using systematic approaches; (f) identify and develop problem solutions in a logical manner; (g) learn independently new technologies; (h) solve complex problems in groups and develop group work. **Alignment of Programme Outcomes:** Programme Outcome 4: This subject contributes to developing critical thinking skills through lectures, in which students learn how to tackle simple problems; assignments and exercises in which they will practice their skills, and quizzes and examination questions in which their skills will be assessed.

	Programme Outcome 5: This subject contributes to problem solving with programming skills through lectures which will teach the students the necessary technical skills involved, assignments and projects in which students will practice these skills, and quizzes and examination questions in which these skills will be assessed. Programme Outcome 7: This subject contributes to team work with group-based projects for students to practice team spirit.
Subject Synopsis/ Indicative Syllabus	

Topic	Duration	
	Lecture	**Lab**
1. Fundamentals of Computing Basic concepts of computers and computing, compilation and interpretation, elementary programming constructs.	6	4
2. Flow controls Basic flow control: selection, repetition and functions.	12	8
3. Data Collections Structures, lists, sets and strings	9	6
4. Program Design Problem solving, problem correctness, testing and debugging	10	7
5. Introduction to Programming Paradigms and Applications Object-based Programming, Functional Programming, Logical Programming.	5	3
Total	**42**	**28**

Teaching/Learning Methodology	This subject emphasizes both the conceptual elements in computer programming and practical experiences. The lectures will be taught in a workshop mode with hands-on exercises reinforcing taught concepts. Students are required to attend the laboratory sessions, which allows them to consolidate their concepts learnt in the lectures. Other practical work helps to reinforce the programming skills learned for applications.

Assessment Methods in Alignment with Intended Learning Outcomes

Specific assessment methods/tasks	% weighting	Intended subject learning outcomes to be assessed (Please tick as appropriate)							
		a	b	c	d	e	f	g	h
1. Assignments	60%	✓	✓			✓	✓		
2. Project(s)		✓	✓			✓	✓		✓
3. Quizzes		✓	✓	✓	✓	✓	✓	✓	✓
4. Examination	40%	✓	✓	✓	✓	✓	✓	✓	
Total	100 %								

Note: Students must pass both the continuous assessment and examination sections to pass the course.

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

The continuous assessment and the final examination will be designed to assess the specified learning outcomes. The formats may include written questions, programming exercises and quizzes.

Student Study Effort Required

Class contact:	
▪ Lecture	42 Hrs.
▪ Lab	14 Hrs.
Other student study effort:	
▪ Assignments, quizzes, projects, exams	64 Hrs.
Total student study effort	120 Hrs.

Reading List and References

(1) John Zelle, Python Programming: An introduction to Computer Science, Franklin, Beedle & Associates, 2004
(2) C. Thomas Wu, An Introduction to Object-Oriented Programming with Java, McGraw-Hill, 3rd Edition Update, 2004.
(3) Deitel & Deitel, Java: How to Program, Prentice-Hall, 6th Edition, 2005.
(4) Deitel & Deitel, C++: How to Program, Prentice-Hall, 6th Edition, 2007.
(5) Patrick Winston, On to C++, Addison-Wesley, 1994