Subject Description Form

Subject Code	COMP447
Subject Title	Scientific Computing
Credit Value	3
Level	4
Pre-requisite / Co-requisite/ Exclusion	Pre-requisite: COMP211 (not applicable for 61025), COMP305 Co-requisite/Exclusion: Nil
Objectives	• To provide an introductory survey of fundamental concepts in scientific computing. • To demonstrate how scientific computing solves scientific and engineering problems. • To equip students with sound skills in solving problems in scientific computing using a scientific computing programming language (e.g. MATLAB) or a professional technical programming language (e.g. C++).
Intended Learning Outcomes	Upon completion of the subject, students will be able to: *Professional/academic knowledge and skills* (a) Understand and apply the basic concepts of scientific computing; (b) Appreciate the limitations of different scientific computing techniques; (c) Write programs to implement scientific computing techniques; (d) Familiarize with a programming environment that supports scientific computing; (e) Be proficient in using the programming constructs for scientific computing; *Attributes for all-roundedness* (f) develop skills in problem solving using systematic approaches; (g) solve complex problems in groups and develop group work. **Alignment of Programme Outcomes:** Programme Outcome 1: This subject contributes to having students to practice their writing skill with project document and report writing. Programme Outcome 4: This subject contributes to developing student critical thinking through tutorials on solving problems. They will practice more in doing

	their project. Programme Outcome 5: This subject contributes to teaching the technical problem solving ability by examples in lectures, practicing such ability in tutorials and measuring such ability by administering quiz or classwork Programme Outcome 6: This subject contributes to informing students about the advancement of scientific computing via lectures or tutorials. Programme Outcome 7: This subject contributes to team work with group-based project for students to practice team spirit.
Subject Synopsis/ Indicative Syllabus	**Topic** **1. Scientific computing fundamentals** Scientific computing overview; mathematical modeling; well-posed problems; approximations in scientific computing; error bounds; stability issues; computer arithmetic; time-space complexity. **2. Scientific computing software** Mathematical software libraries; scientific computing environments. **3. Numerical linear algebra** Solving linear equations; norms and condition numbers; orthogonal methods to solve linear least squares; eigenvalues and singular values; pseudo-inverses. **4. Nonlinear systems** Nonlinear equations in one dimension; systems of nonlinear equations. **5. Optimization** One-dimensional optimization; multidimensional unconstrained optimization; nonlinear least squares; constrained optimization. **6. Integration and differential equations** Finite difference approximations, automatic differentiation; initial value problems for ordinary differential equations; boundary value problems for ordinary equations; finite numerical quadratures; double and multiple integrals. **7. Partial differentiation** Time-dependent problems; time-independent problems; direct methods for sparse linear systems; iterative methods. **Laboratory Experiment:** **Topic** 1. Programming for scientific computing (e.g. vector computation, matrix computation, etc.). 2. Solving scientific problems using scientific computing techniques. 3. Solving engineering problems using scientific computing techniques.

Teaching/Learning Methodology

Teaching is based on lectures which include solving technical problems in scientific computing (aligned to Programme Outcomes 5, 6). Tutorials are used to provide examples of problems and to show how solutions are developed (aligned to Programme Outcomes 4, 6). Quizzes and/or classworks are administered to students to strength their technical problem solving ability (aligned to Programme Outcome 5). There is a project that students need to write their report (aligned to Programme Outcomes 1,4). This project is typically a group project (aligned to Programme Outcome 7).

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
1. Assignments								
2. Lab exercises	10%	✓			✓	✓		
3. Project	25%	✓		✓			✓	✓
4. Mid-term	20%	✓	✓					
5. Examination	45%	✓	✓	✓				
Total	100 %							

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

The project tests whether the student can understand and apply the basic concepts of scientific computing, by requiring them to write programs implementing scientific computing techniques. In doing so, this develops skills in problem solving. The project is group work. Laboratory exercises help students to understand and apply basic concepts, familiarize with a programming environment for scientific computing and be proficient in using programming constructs. The mid-terms tests the basic concepts learnt by the students and the limitations understood by the students. The examination tests these as well as the ability to write programs implementing scientific computing techniques.

Student Study Effort Expected

Class contact:	
▪ Lecture	39 Hrs.
▪ Laboratory	0 Hrs.
Other student study effort:	
▪ Project	14 Hrs.
▪ Self Study	17 Hrs.
Total student study effort	70 Hrs.

Reading List and References

Reference Books:

1. M.T. Health, Scientific Computing: An Introductory Survey, 2nd ed., McGraw-

	Hill, 2002. 2. A. Quateroni and F. Saleri, Scientific Computing with MATLAB (Text in Computational Science and Engineering 2), Springer-Verlag, 2003. 3. J.S. Liu, Monte Carlo Strategies in Scientific Computing, Springer, 2002. 4. B. Lucqion and O. Pironneau, Introduction to Scientific Computing, John Wiley & Sons, 1998. 5. G.H. Golub and J.M. Ortega, Scientific Computing and Differential Equations: An Introduction to Numerical Methods, Academic Press, 1992.