Subject Description Form

Subject Code	COMP307
Subject Title	Operating Systems and System Programming
Credit Value	3
Level	3
Pre-requisite / Co-requisite/ Exclusion	Nil
Objectives	This subject provides students knowledge on: • basic concepts of operating system and the abilities to maintain it; • writing system software with the aid of sophisticated system programming tools.
Intended Learning Outcomes	Upon completion of the subject, students will be able to: *Professional/academic knowledge and skills* (a) identify the services provided by operating systems. (b) understand the internal structure of an operating system and be able to write programs using system calls; (c) understand and solve problems involving process control, mutual exclusion, deadlock and synchronization. *Attributes for all-roundedness* (d) Students need to solve complex problems in groups. This will develop their skills in problem solving using systematic approaches; (e) build up on team spirit, presentation and technical writing skills. **Alignment of Programme Outcomes:** Programme Outcome 1: This subject contributes to having students practice their writing skills with project document and report writing. Programme Outcome 2: This subject contributes to teaching students to understand how operating systems can affect the computing systems. Programme Outcome 4: This subject contributes to developing student critical thinking through tutorial and lab exercises on solving problems. They will also practice more in written assignments, programming exercises, and project. Programme Outcome 5: This subject contributes to problem solving with

<table>
<tr><td></td><td>programming skills through lab exercises and project with proper design and implementation.

Programme Outcome 7: This subject contributes to team work with group-based project for students to build team spirit and practice collaboration skills.</td></tr>
<tr><td>Subject Synopsis/ Indicative Syllabus</td><td>
<table>
<tr><th>Topic</th></tr>
<tr><td>1. Introduction
Operating system concept; operating system structure; machine and run-time environment; concept of system programming.</td></tr>
<tr><td>2. Process management and coordination
Process concepts; process manipulation; concurrent processes; inter-process communication, mutual exclusion; synchronization; deadlock; scheduling algorithms.</td></tr>
<tr><td>3. Memory management
Memory allocation; Paging and segmentation; virtual memory.</td></tr>
<tr><td>4. File management and secondary storage management
File system structure; file system implementation; secondary storage allocation; disk scheduling.</td></tr>
<tr><td>5. I/O subsystem
I/O devices; device controllers; device drivers.</td></tr>
<tr><td>6. Interrupt processing
Interrupt mechanism; signal processing.</td></tr>
<tr><td>7. Protection and Security
Concepts of computer protection and security.</td></tr>
</table>

Laboratory Experiment:
• Hands-on experience on Unix or Linux.
• Shell, pipe and script.

Case Study:

Unix, Windows NT, Linux.
</td></tr>
<tr><td>Teaching/Learning Methodology</td><td>This subject consists mainly of class lectures and laboratory exercises. Students may learn from case studies as well.</td></tr>
</table>

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	
1. Assignments	55%		✓	✓	✓		
2. Lab exercises			✓		✓		
3. Project		✓	✓	✓	✓	✓	
4. Mid-term		✓	✓	✓	✓		
5. Quizzes		✓	✓	✓	✓		
6. Examination	45%	✓	✓	✓	✓		
Total	100 %						

Student Study Effort Expected

Class contact:	
▪ Lecture	26 Hrs.
▪ Tutorial/Lab	13 Hrs.
Other student study effort:	
▪	Hrs.
▪ Total student study effort	39 Hrs.

Reading List and References

Textbooks:

1. Silberschatz, A. and Galvin, P., Operating System Concepts, 7/E, John Wiley and Sons, 2006.

Reference Books:

1. Stallings, W., Operating Systems: Internals and Design Principles, 5/E, Prentice Hall, 2005.
2. Keith Haviland, Dina gray, Ben Salama, Unix System Programming : A Programmer's Guide to Software Development, Second Edition, Addison Wesley, 1999.
3. Gary Nutt, Operating Systems: A Modern Perspective, 3/E, Addison Wesley, 2004.
4. Aho, A.V., Sethi, R. and Ullman, J.D., Compilers Principles, Techniques and Tools, Addison Wesley, 1986.