

Module Title:	Operating Systems
Language of Instruction:	English
Credits:	10
NFQ Level:	6
Module Delivered In	8 programme(s)
Teaching & Learning Strategies:	Learning is divided into lecture and practical sessions over one semester. The practical sessions will provide students with hands on experience in installing, using, configuring and managing an operating system. It will also provide the opportunity to implement and reinforce material presented in lectures, to learn by doing.
Module Aim:	To provide students with practical skills and working knowledge of an industry standard operating system.
Learning Outcomes	
<i>On successful completion of this module the learner should be able to:</i>	
LO1	Install and configure an operating system with appropriate security considerations.
LO2	Demonstrate practical skills in the operation and management of an operating system.
LO3	Explain the operation of the process and memory manager of an operating system.
LO4	Describe the relationship between a file system, file manager and device manager.
LO5	Automate simple tasks by using commands to write and execute scripts.
Pre-requisite learning	
Module Recommendations <i>This is prior learning (or a practical skill) that is recommended before enrolment in this module.</i>	
No recommendations listed	
Incompatible Modules <i>These are modules which have learning outcomes that are too similar to the learning outcomes of this module.</i>	
No incompatible modules listed	
Co-requisite Modules	
No Co-requisite modules listed	
Requirements <i>This is prior learning (or a practical skill) that is mandatory before enrolment in this module is allowed.</i>	
No requirements listed	

Module Content & Assessment

Indicative Content

Introduction to Operating Systems

Role of an OS as the system software, types of OS. Examples of OS in use (server, desktop, mobile, embedded etc.) Examination of Linux and Windows as a desktop OS. Four managers of an OS (File, Process, Memory and Device).

Installing an OS [Practical]

Installation and configuration of an OS. Implementation of typical security features. Review of typical devices on a system and viewing the boot process. Command line access and basic commands.

File Management

Files, directories and file systems. Absolute and relative paths. Relationship between file management and disk management. Storage devices and examination of the storage/memory hierarchy.

File Systems [Practical]

Typical hierarchical directory structure, navigating the file system, managing files and directories Creating and editing files via command line. File compression and archiving. File and disk system utilities.

Computer Security

Introduction to computer security and the role an OS plays in securing systems. Threats, vulnerabilities and types of breaches. The CIA model and layering of host, network and application security.

OS Security [Practical]

Typical OS security practices. Managing file and directory permissions on a system. User and group management on a system. Configuring firewall rules and examination of system log files.

Process Management

What is a process? Explanation of process vs a program. A process as a unit of work in a system and the process lifecycle. Process control block, context switches and scheduling. Deadlocks and starvation.

Process Management [Practical]

Using commands to access process information, manage processes by starting, stopping, and resuming them. Viewing process resource usage, comparison of process and task managers in various OS.

Memory Management

Evolution from fixed and variable partitions to paged/segmented systems. Physical vs logical address space, implementing virtual memory. Trade off's involved when implementing memory management schemes.

Data Manipulation [Practical]

Working with text files. Exploring advanced command line features such as using variables and quoting. Redirecting input and output streams, using regular expressions and filtering text.

Choosing an OS

Comparison of contemporary Operating Systems on the market. Considerations to include; suitability for role, ease of use, hardware resource requirements and use of proprietary technology.

Scripting [Practical]

Using an editor to create basic scripts to automate standard user tasks. Using basic shell commands, variables and control statements.

Assessment Breakdown	%
Continuous Assessment	50.00%
Practical	10.00%
End of Module Formal Examination	40.00%

Continuous Assessment

Assessment Type	Assessment Description	Outcome addressed	% of total	Assessment Date
Multiple Choice Questions	Diagnostic quiz	3	20.00	Week 5
Practical/Skills Evaluation	Lab exam	2,5	30.00	Week 11

No Project

Practical

Assessment Type	Assessment Description	Outcome addressed	% of total	Assessment Date
Practical/Skills Evaluation	Participation and completion of practical lab tasks	1,2	10.00	Every Week

End of Module Formal Examination

Assessment Type	Assessment Description	Outcome addressed	% of total	Assessment Date
Formal Exam	End of semester exam	3,4	40.00	End-of-Semester

Module Workload

Workload: Full Time		
<i>Workload Type</i>	<i>Frequency</i>	<i>Average Weekly Learner Workload</i>
Lecture	12 Weeks per Stage	2.00
Laboratory	12 Weeks per Stage	4.00
Independent Learning	15 Weeks per Stage	11.87
Total Hours		250.00

Module Delivered In

Programme Code	Programme	Semester	Delivery
CW_KWCCD_B	Bachelor of Science (Honours) in Creative Computing and Digital Innovation	2	Mandatory
CW_KCCYB_B	Bachelor of Science (Honours) in Cyber Crime and IT Security	2	Mandatory
CW_KCCIT_B	Bachelor of Science (Honours) in Information Technology Management	2	Mandatory
CW_KCSOF_B	Bachelor of Science (Honours) in Software Development	2	Mandatory
CW_KCCYB_D	Bachelor of Science in Cybercrime and IT Security	2	Mandatory
CW_KCCSY_D	Bachelor of Science in Information Technology Management	2	Mandatory
CW_KCSOF_D	Bachelor of Science in Software Development	2	Mandatory
CW_KCCOM_C	Higher Certificate in Science in Computing Programming	2	Mandatory