



Module Specification

Computer Networks and Operating Systems

Version: 2023-24, v2.0, 03 Aug 2023

Contents

Module Specification	1
Part 1: Information	2
Part 2: Description	2
Part 3: Teaching and learning methods	4
Part 4: Assessment.....	5
Part 5: Contributes towards	6

Part 1: Information

Module title: Computer Networks and Operating Systems

Module code: UFCFQ4-30-2

Level: Level 5

For implementation from: 2023-24

UWE credit rating: 30

ECTS credit rating: 15

College: Faculty of Environment & Technology

School: FET Dept of Computer Sci & Creative Tech

Partner institutions: None

Field: Computer Science and Creative Technologies

Module type: Module

Pre-requisites: Computer and Network Systems 2023-24

Excluded combinations: None

Co-requisites: None

Continuing professional development: No

Professional, statutory or regulatory body requirements: None

Part 2: Description

Overview: Pre-requisites: students must take UFCF93-30-1 Computer and Network Systems and UFCFF6-30-1 Programming in C or UFCFC3-30-1 Introduction to OO Systems Development.

Features: Not applicable

Educational aims: See Learning Outcomes

Outline syllabus: Operating Systems:

Operating System Organisation models and structures

History and implications of using Open Source code in operating systems. Licensing issues and their legal implications.

Process and Object Management kernel services, interrupt handlers, scheduling.

Inter-process Communication event handling, message passing, synchronous-asynchronous, shared memory.

Concurrency and Synchronisation semaphores, critical regions, monitors, message passing, multi-threaded processes.

Memory Management Organisation algorithms and policies, Virtual Memory Management.

I/O Management Device driver design, Buffering and interrupt handling.

File and Persistent Object Management File organisation, directories and naming, index nodes, disk block management. Network and distributed file systems.

Protection and Security Models for secure computing, access control, capability based systems, access control lists.

Virtualisation. History of VMs. Variety of virtualisation – full, partial, para. Emulators, simulators and virtualisation. VM in languages – Java VM.

Embedded and mobile OS. Background to embedded and mobile os. Hardware and software requirements for embedded/mobile. Embedded/mobile OS.

Networked Systems:

Computer network architecture's and models. Layered models, peer protocols, the ISO OSI model.

Protocol Specification and Design Specification techniques -FSM, layered protocols, error correction Connection vs connectionless protocols.

Medium Access Control Protocols MAC techniques.

Subnetworks and Internetworks network layer design, routing and switching, addressing and naming network topology.

Transport Services TLIs.

Network and Distributed Systems Management Security issues, fault, monitoring and accounting issues.

TCP/IP protocols IP layer, ICMP, ARP TCP socket programming Applications IPV4 and IPng Administering a TCP IP network.

System Administration Specifying and installing an OS and network Initialise the system for user and applications Install devices, software packages and communication links. Making the system secure, investigation of security strategies Instigation of system maintenance - backup, user control Document system and system modifications

Part 3: Teaching and learning methods

Teaching and learning methods: For the most part the course will be delivered through practicals and lectures. The theoretical content will be covered in lectures. In the practical sessions students will gain understanding through designing and implementing system software components. For the systems administration part of the course the students will be divided into groups and be required to configure and manage a computer system and offer this computing service to an acceptable professional standard to the rest of the cohort.

An extended case-study, supported by focussed tutorials and practicals, will allow

the students to follow through an example application from design to implementation, and appreciate the relevance of all the component parts of the module syllabus.

Examples of case studies could be: creating a VPN (virtual private network) within the existing network, implementing a firewall/bastion host/proxy security strategy within the network, full automate the user account administration of the system.

Module Learning outcomes: On successful completion of this module students will achieve the following learning outcomes.

MO1 Show a detailed knowledge and understanding of the design, structure and implementation of modern networked operating systems (NOS) as well as the data structures and interfaces of a NOS

MO2 Write small utility programs, in both script and compiler level languages, that interface to the system primitives

MO3 Build, install, boot, administer and manage a NOS and manage the users on the system

MO4 Understand the networking protocols that underpin the operation of a NOS

MO5 Understand the security problems and solutions on a network

Hours to be allocated: 300

Contact hours:

Independent study/self-guided study = 228 hours

Face-to-face learning = 72 hours

Total = 300

Reading list: The reading list for this module can be accessed at [readinglists.uwe.ac.uk](https://uwe.rl.talis.com/modules/ufcfq4-30-2.html) via the following link <https://uwe.rl.talis.com/modules/ufcfq4-30-2.html>

Part 4: Assessment

Assessment strategy: At both first sit and resit, students will be assessed through a portfolio. The portfolio will consist of a mix of practical worksheets, set assignment

tasks and answers to theoretical questions. The worksheets are designed to ensure sustained student engagement whilst the assignment tasks allow the student to demonstrate their mastery of a number of practical skills. The final portfolio task requires the consolidation of the complete module material.

The more theoretical aspects of the course are assessed through answers to theoretical questions related to the coursework.

Assessment tasks:

Portfolio (First Sit)

Description: Portfolio of practical tasks.

Weighting: 100 %

Final assessment: Yes

Group work: No

Learning outcomes tested: MO1, MO2, MO3, MO4, MO5

Portfolio (Resit)

Description: Portfolio of practical tasks.

Weighting: 100 %

Final assessment: Yes

Group work: No

Learning outcomes tested: MO1, MO2, MO3, MO4, MO5

Part 5: Contributes towards

This module contributes towards the following programmes of study:

Software Engineering {Dual} [Taylors] BSc (Hons) 2022-23

Software Engineering [Sep][FT][Frenchay][3yrs] - Not Running BSc (Hons) 2022-23

Software Engineering {Dual} [Aug][FT][Taylors][3yrs] BSc (Hons) 2022-23

Software Engineering [Sep][SW][Frenchay][4yrs] - Not Running BSc (Hons) 2022-23

Computing {Dual} [Aug][FT][Taylors][3yrs] - Not Running BSc (Hons) 2022-23

Computing {Dual} [Mar][FT][Taylors][3yrs] - Not Running BSc (Hons) 2022-23

Computing {Dual} [Mar][SW][Taylors][4yrs] - Not Running BSc (Hons) 2022-23

Computing {Dual} [Aug][SW][Taylors][4yrs] - Not Running BSc (Hons) 2022-23

Software Engineering {Dual} [Mar][FT][Taylors][3yrs] BSc (Hons) 2022-23

Software Engineering [Jan][FT][Northshore][3yrs] - Not Running BSc (Hons) 2022-23

Software Engineering {Foundation} [Feb][FT][GCET][4yrs] - Withdrawn BEng (Hons)
2021-22

Software Engineering {Foundation} [Oct][FT][GCET][4yrs] - Withdrawn BEng (Hons)
2021-22