STUDENT AND ACADEMIC SERVICES

MODULE SPECIFICATION

Part 1: Information			
Module Title	Computer and Network Systems		
Module Code	UFCF93-30-1	Level	Level 4
For implementation from	2018-19		
UWE Credit Rating	30	ECTS Credit Rating	15
Faculty	Faculty of Environment & Technology	Field	Computer Science and Creative Technologies
Department	FET Dept of Computer Sci & Creative Tech		
Contributes towards	Computing {Dual} [Aug][SW][Taylors][4yrs] BSc (Hons) 2018-19 Computer Science [Sep][SW][Frenchay][4yrs] BSc (Hons) 2018-19 Software Engineering [Sep][SW][Frenchay][4yrs] BSc (Hons) 2018-19 Forensic Computing and Security [Sep][SW][Frenchay][4yrs] BSc (Hons) 2018-19 Software Engineering [Jan][FT][Northshore][3yrs] BSc (Hons) 2018-19 Software Engineering {Dual} [Aug][FT][Taylors][3yrs] BSc (Hons) 2018-19 Software Engineering {Dual} [Mar][FT][Taylors][3yrs] BSc (Hons) 2018-19 Software Engineering [Sep][FT][Frenchay][3yrs] BSc (Hons) 2018-19 Computer Science [May][FT][Villa][3yrs] BSc (Hons) 2018-19 Computer Science [Jan][FT][Villa][3yrs] BSc (Hons) 2018-19 Computer Science [Sep][FT][Villa][3yrs] BSc (Hons) 2018-19 Computer Science [Sep][FT][Frenchay][3yrs] BSc (Hons) 2018-19 Forensic Computing and Security {Dual} [Mar][FT][Taylors][3yrs] BSc (Hons) 2018-19 Forensic Computing and Security {Dual} [Aug][FT][Taylors][3yrs] BSc (Hons) 2018-19 Computing {Dual} [Mar][FT][Taylors][3yrs] BSc (Hons) 2018-19 Forensic Computing and Security [Sep][SW][Frenchay][4yrs] BSc (Hons) 2018-19 Computing {Dual} [Aug][FT][Taylors][3yrs] BSc (Hons) 2018-19 Computing {Dual} [Mar][SW][Taylors][4yrs] BSc (Hons) 2018-19		
Module type:	Standard		
Pre-requisites	None		

Excluded Combinations	None
Co- requisites	None
Module Entry requirements	None

Part 2: Description

Educational Aims: See Learning Outcomes

Outline Syllabus: This module seeks to introduce the concepts of computer hardware, operating systems, programming and networking.

Computer Hardware:

The Principal Functional Units and the Fetch-execute cycle
Interrupts
Numbers and Logic circuits
Memory circuits
Adding, Subtraction, Multiplication, Division circuits.

Operating Systems:

Memory Management and Scheduling
Processes and Threads
Introduction to Linux
Comparing Windows and Linux
Caching

Programming:

Languages and Compilers
Different ways to programme
Finite State Machine
Using Threads and Semaphores
Data Structures

Networking:

LAN and WAN
Cell Phone Network
Client Server
Security Problems
Security Solutions

Rounding up:

The move to low power consumption and sustainability
Pulling it all together

Teaching and Learning Methods: The module is delivered through weekly and fortnightly lectures and weekly lab sessions. Each lecture will direct the course and introduce the new ideas and skills required. Then small group lab sessions will enable each student to carry out the practical exercises described in the associated work-sheet under the guidance of a Lab Tutor. Highly effective PAL tutoring sessions are provided to support students every week.

The teaching material is available from Blackboard. A course text is also recommended.

Scheduled learning includes lectures, tutorials, practical lab classes, and PAL mentoring

Independent learning includes time engaged with essential reading, assignment coursework and self-assessment tests.

This module will involve 9 hours contact time per fortnight (3 hours of lectures, 4 hours practical and 2 hours of PAL sessions).

Activity (hrs)
Contact time (108)
Assimilation and development of knowledge (117)
Exam preparation (37.5)
Coursework preparation (37.5)
Total study time (300)

Part 3: Assessment

The assessment is split 50/50 between practical coursework and tests. Component B, coursework normally involves the production of software to implement a specification, coupled with a report on its implications. The actual assignment topics are carefully chosen to demonstrate some basic principles which are especially significant to the course. For example, data transmissions flow, error control, multi-tasking, and the use of FSDs or runtime debugging. There is a library exercise which gives students practice in using research techniques for use in their reports.

The programming will be challenging for most of the students and must be demonstrated and explained orally to a tutor for part of the assessment. In this way, students develop the skill and confidence to talk about the subtle intricacies of their software, and so become aware and proud of their achievements.

All the coursework is required to be carried out individually but team working will be allowed in the second assignment to encourage communication.

More often than not, the coursework will contain proven source code to assist students to start the assignment. This in itself is a considerable challenge because reading other's code is not a facile accomplishment.

The component A grade will be obtained from two tests. By offering within-course tests, helpful feedback can be delivered, and students can work to improve their final grade. Tests will be delivered in a multiple choice format, with the marked scripts returned quickly to the students for immediate review.

First Sit Components	Final Assessment	Element weighting	Description
Set Exercise - Component B		5 %	Library Exercise
Report - Component B		22.5 %	Programming assignment with report (800 words)
Report - Component B	✓	22.5 %	Programming assignment with report (800 words)
Examination - Component A		25 %	Written test (2 hours)
Examination - Component A		25 %	Written test (2 hours)
Resit Components	**Final Assessment**	**Element weighting**	**Description**
Set Exercise - Component B	✓	50 %	Programming assignment with report (1600 words)
Examination - Component A		50 %	Written examination (3 hours)

Part 4: Teaching and Learning Methods

Learning Outcomes

On successful completion of this module students will be able to:

	Module Learning Outcomes
MO1	Demonstrate detailed knowledge and understanding of the structure and function of modern computer systems
MO2	Apply fundamental principles of combinatorial digital logic to expose the principal building blocks of computer systems
MO3	Recognise that computers can be viewed a hierarchy of functional layers, and understand the close interplay of hardware and software
MO4	Compare features of new computer architectures with the original von Neumann model.
MO5	Understand the important role of an Operating System
MO6	Write programs using assembler
MO7	Use numeric and character data typing and convert between them
MO8	Understand some of the technical principles and practical details of computer networking
MO9	Conduct research into the impact on society of decisions related to sustainability, e.g. decisions on power consumption by computing devices

Contact Hours

Contact Hours	
Independent Study Hours:	
Independent study/self-guided study	192
Total Independent Study Hours:	192
Scheduled Learning and Teaching Hours:	
Face-to-face learning	108
Total Scheduled Learning and Teaching Hours:	108
Hours to be allocated	300
Allocated Hours	300

Reading List

The reading list for this module can be accessed via the following link:

https://uwe.rl.talis.com/index.html