STUDENT AND ACADEMIC SERVICES

MODULE SPECIFICATION

Part 1: Information

Module Title	Program Design and Implementation		
Module Code	UFCEXX-30-0	Level	Level 3
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	Computer Security and Forensics [Oct][FT][GCET][4yrs] BSc (Hons) 2018-19 Automation and Robotics Engineering {Foundation} [Feb][FT][GCET][4yrs] BEng (Hons) 2018-19 Software Engineering [Oct][FT][GCET][4yrs] BEng (Hons) 2018-19 Computer Security and Forensics {Foundation} [Sep] [FT] [GCET] [4yrs] BSc (Hons) 2018-19 Electronics and Telecommunication Engineering [Feb][FT][GCET][4yrs] BEng (Hons) 2018-19 Mechanical Engineering and Vehicle Technology [Feb][FT][GCET][4yrs] BEng (Hons) 2018-19 Computer Security and Forensics [Feb][FT][GCET][4yrs] BSc (Hons) 2018-19 Mechanical Engineering and Vehicle Technology [Oct][FT][GCET][4yrs] BEng (Hons) 2018-19 Electronics and Telecommunication Engineering [Oct][FT][GCET][4yrs] BEng (Hons) 2018-19 Automation and Robotics Engineering {Foundation} [Oct][FT][GCET][4yrs] BEng (Hons) 2018-19 Instrumentation and Control Engineering {Foundation} [Feb][FT][GCET][4yrs] BEng (Hons) 2018-19 Multimedia Technology [Oct][FT][GCET][4yrs] - Not Running BSc (Hons) 2017-18 Software Engineering [Feb][FT][GCET][4yrs] BEng (Hons) 2018-19 Instrumentation and Control Engineering {Foundation} [Oct][FT][GCET][4yrs] BEng (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: Basic programming concepts:

Sequence, selection and iteration constructs; Simple data types, structured data types: arrays and structs; Access scope of local and global variables; The use of functions with parameters; modularisation: coupling and cohesion and structure charts; Arithmetic and logical operations; Handling strings; Use of pointers to access arrays and structs

Data i/o using keyboard, screen and web; Simple file handling; Operations: searching and sorting

Familiarisation with IDE (integrated Development Environment) and command line operation; Testing; Proof of testing

Polyas approach to problem solving; Algorithmic and strategy problems such as river crossing and NIM

Software design and development:

Requirements analysis, functional and non-functional requirements
Design and development techniques; Top-down vs bottom-up; Functional decomposition; Iterative design, prototyping
Implementation; Testing, verification and validation

Teaching and Learning Methods: This module will use lectures to introduce new concepts and direct the students to texts and web sources, while associated practical laboratory sessions will expose and explore the material in greater depth. Students will be expected to carry out independent study in parallel with the timetabled periods.

The design and production of programs requires practice if it is to be mastered. This module will include the setting and solving of a series of small problems, each introducing a new idea or technique to be mastered. Some of these will be solved in preparation for classes that will discuss the merits or otherwise of certain approaches. There will be regular feedback on the exercises to facilitate the development of skills as the module progresses. Lectures will be used to introduce new ideas.

Scheduled learning includes lectures, seminars, tutorials, project supervision, demonstration, practical classes and workshops; fieldwork; external visits; work based learning; supervised time in studio/workshop.

Independent learning includes hours engaged with essential reading, case study preparation, assignment preparation and completion etc. These sessions constitute an average time per level as indicated in the table below. Scheduled sessions may vary slightly depending on the module choices you make.

Placement learning: may include a practice placement, other placement, year abroad.

Activity (hrs)
Contact time (72)
Assimilation and development of knowledge (148)
Coursework preparation (80)
Total study time (300)

Part 3: Assessment

The assessment will consist of:

1. A series of workshop exercises of increasing complexity aimed at developing competency and confidence in the use of a programming environment and basic programming skills.

2. An in-class test conducted early in the delivery to test conceptual understanding and provide early feedback.

3. A final written examination, testing depth of understanding and evaluative skills

First Sit Components	Final Assessment	Element weighting	Description
Portfolio - Component B		60 %	Portfolio of programming exercises
In-class test - Component A		16 %	In-class tests
Examination - Component A	✓	24 %	Examination
Resit Components	**Final Assessment**	**Element weighting**	**Description**
Set Exercise - Component B		60 %	Set of programming exercises
Examination - Component A	✓	40 %	Examination

Part 4: Teaching and Learning Methods

Learning Outcomes

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

	Module Learning Outcomes
MO1	Demonstrate understanding, and appropriate use, of a variety of notations for the specification of processing rules and algorithms
MO2	Discuss the relative merits of different programming languages and their use in the development of software for different applications and platforms
MO3	Demonstrate understanding of the functional role of common software development tools and the use of libraries in the development of software
MO4	Demonstrate understanding of the structure and syntax of a high level programming language and the use of data structures and syntactic constructs in the implementation of algorithms
MO5	Discuss the principles of good design and apply these in the development and evaluation of program designs
MO6	Use appropriate methods to design, implement and test programs to achieve functional and non-functional requirements, derived from a simple requirements specification

Contact Hours

Contact Hours	
Independent Study Hours:	
Independent study/self-guided study	228
Total Independent Study Hours:	228
Scheduled Learning and Teaching Hours:	
Face-to-face learning	72
Total Scheduled Learning and Teaching Hours:	72
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/modules/ufcexx-30-0.html