MODULE SPECIFICATION

Part 1: Information			
Module Title	Principles of Computing		
Module Code	UFCFA3-30-1	Level	Level 4
For implementation from	2019-20		
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		
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: Computation models: Finite Machines; Pushdown Automata; Turing Machines. How these abstract machines work. What limitations they have. How do apply them to real world applications. The significance of the Universal Turing Machines.

Formal Languages: words, sentences, languages, grammars, productions. Links to computing models. How to formally define languages. How a compiler detects syntax errors.

Algorithms: Classes of algorithms, search algorithms and sorting algorithms. Time and space complexity of algorithms. NP-complete problems.

Recursion: Inductive definitions and recursive programs.

Logic: Propositional and Predicate logic. Truth tables for basic logic operators. Inference methods.

Mathematical Structures: Numbers. Sets. Functions. Relations. Matrices. Application of mathematical structure to computing. Enumerating (counting) these structures.

Graph Theory: Theory and its applications as a modelling tool. Classical problems: finding the shortest route on a graph and the travelling salesman problem.

Teaching and Learning Methods: Scheduled learning:

This module will be delivered by a series of lectures accompanied by a mixture of tutorials and hands-on practicals. The practicals will use software tools to help students understand the course contents. The tutorials will use the course handouts as the basis of discussions of issues presented in the lectures

Independent learning:

Students are expected to work outside scheduled classes on practice, assignment work and directed reading as described below.

This module will involve 6 hours contact time per fortnight. The time will be more or less equally divided between lecture sessions and laboratory sessions.

Activity (hrs)
Contact time (72)
Assimilation and development of knowledge (148)
Exam preparation (40)
Coursework preparation (40)
Total study time (300)

Part 3: Assessment

Assessment consists of one examination (component A) and two coursework (component B).

The examination takes place at the end of academic year (worth 50%).

The coursework will consist of two elements:

E-assessment – for mathematical questions with a short and unique answer

Written-assessment – for other types tasks which cannot be marked automatically

The written assessment element will be a portfolio of tasks which the students will complete throughout the year. This strategy helps students to consolidate the taught content immediately after lectures.

Note that, e-assessment is not used for resit.

First Sit Components	**Final Assessment**	**Element weighting**	**Description**
Written Assignment - Component B		25 %	Written Assessment – a portfolio of tasks related to problems of computational theory
Set Exercise - Component A		25 %	e-Assessment – short answers to questions in mathematics
Examination - Component A	✓	50 %	Examination (2 hours)

Resit Components	Final Assessment	Element weighting	Description
Written Assignment - Component B		50 %	Written Assessment – a combination of tasks related to mathematics and problems of computational theory.
Examination - Component A	✓	50 %	Examination (2 hours)

Part 4: Teaching and Learning Methods

Learning Outcomes

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

Module Learning Outcomes	Reference
Understand simple models of computation and formulate small problems in terms of those models	MO1
Define the syntax of formal languages in terms of productions. Define functions using recursion	MO2
Explain algorithmic behaviour of programs in appropriate formal terms and Big-O notation	MO3
Design and simulate abstract computation models: Finite Automata, Push Down Automata, and Turing Machines.	MO4
Appreciate the limitations of computers	MO5
Use mathematical language, notation and methods in the description and analysis of problems in appropriate areas of application within computing.	MO6
Begin to abstract general principles from studying particular problems and solutions	MO7
Recognise the fundamental role of foundation mathematics and discrete mathematical structures within computing	MO8

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/ufcfa3-30-1.html

Part 5: Contributes Towards
This module contributes towards the following programmes of study: Computing [Sep][FT][Frenchay][3yrs] BSc (Hons) 2019-20 Software Engineering for Business [Sep][FT][Frenchay][3yrs] BSc (Hons) 2019-20 Software Engineering for Business [Sep][SW][Frenchay][4yrs] BSc (Hons) 2019-20 Computing [Sep][SW][Frenchay][4yrs] BSc (Hons) 2019-20 Computing {Dual} [Aug][FT][Taylors][3yrs] BSc (Hons) 2019-20 Computing {Dual} [Aug][SW][Taylors][4yrs] BSc (Hons) 2019-20 Computing {Dual} [Mar][FT][Taylors][3yrs] BSc (Hons) 2019-20 Computing {Dual} [Mar][SW][Taylors][4yrs] BSc (Hons) 2019-20 Software Engineering [Oct][FT][GCET][4yrs] BEng (Hons) 2018-19 Computing {Foundation} [Sep][FT][Frenchay][4yrs] BSc (Hons) 2018-19 Computer Science {Foundation} [Sep][SW][Frenchay][5yrs] BSc (Hons) 2018-19 Computer Science {Foundation} [Sep][FT][Frenchay][4yrs] BSc (Hons) 2018-19 Computing {Foundation} [Sep][SW][Frenchay][5yrs] BSc (Hons) 2018-19 Software Engineering [Feb][FT][GCET][4yrs] BEng (Hons) 2018-19 Software Engineering for Business {Foundation} [Sep][FT][Frenchay][4yrs] BSc (Hons) 2018-19 Software Engineering for Business {Foundation} [Sep][SW][Frenchay][5yrs] BSc (Hons) 2018-19