MODULE SPECIFICATION

Part 1: Information			
Module Title	Parallel Computing		
Module Code	UFCFFL-15-M	Level	Level 7
For implementation from	2020-21		
UWE Credit Rating	15	ECTS Credit Rating	7.5
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

Overview: The module will cover the theory and background and state-of-art knowledge of parallel computing field in line with related computing fields such as distributed and high-performance computing extended with cluster and cloud computing.

Educational Aims: The main aim is to develop skills in problem solving using high performance computing infrastructures and state of art technologies for real world computing and engineering problems.

Outline Syllabus: The module content can be outlined in the following syllabus:

Parallel Computing:
Introduction to parallel computing
Introduction to parallel architectures
Parallel programming abstractions, e.g. OpenMP, Actors, TBB

Heterogeneous Computing:
GPUs, DSPs, etc.
Parallel programming abstractions, e.g. OpenCL 4.1, Cuda
Applications, e.g. image processing, HPC

Distributed Computing (overview, to provide context with parallel):
Clusters
High-Performance Computing
Cloud computing

Teaching and Learning Methods: See Syllabus and Assessment.

Part 3: Assessment

The assessment of this module is based on two components; Component A is a condition-controlled assessment via written unseen exam, It will cover the majority of the learning outcomes from LO1 to LO5 focusing on the theory and state-of-art knowledge of parallel computing. The second component, Component B, is a coursework assignment, which will assess the practical skills through demonstration of innovative solutions designed and developed for practical problems along with submission of a logbook.

Component B is a single component, which includes a demonstrable development, (parallel algorithms to solve a given problem), and a logbook to report the stages, progress and performance analysis of the development. Both demonstration and the logbook are separately assessed.

The summative assessment will be achieved based on these two assessment components (Component A and B) while the formative assessment will be provided as oral feedback throughout the laboratory sessions particularly with respect to the design development and the log-book entries.

Resit:
Resit coursework requires reworking the original assignment, which should be improved following the summative feedback from the first sit. The development should be revised to meet the requirements, and the logbook should be re-written accordingly. The demonstration of development will be videoed for submission together with logbook.

First Sit Components	**Final Assessment**	**Element weighting**	**Description**
Portfolio - Component B	✓	60 %	Logbook (2000- 3000 words) and practical demonstration of final product
Examination (Online) - Component A		40 %	Online Written Examination (2 hours) 24 hour window
Resit Components	**Final Assessment**	**Element weighting**	**Description**
Portfolio - Component B	✓	60 %	Logbook (2000- 3000 words) and video demonstration of final product (15 mins)
Examination (Online) - Component A		40 %	Online Written Examination (2 hours) 24 hour window

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**
Distinguish, contrast, and apply the main concepts of sequential, concurrent, and parallel computing	MO1
Be able to critically evaluate and assess the effectiveness of parallel computation in homogenous and heterogeneous environments	MO2
Distinguish, contrast, and reflect between different hardware abstractions for parallelism, e.g. multi-core, many-core, and vector architectures	MO3

	Develop programs for parallel systems, e.g. using OpenMP for single-node, and Cuda for accelerators, and MPI for multi-node	MO4
	Develop parallel designs and algorithm design and implementation	MO5

Contact Hours	**Independent Study Hours:**	
	Independent study/self-guided study	114
	Total Independent Study Hours:	114
	Scheduled Learning and Teaching Hours:	
	Face-to-face learning	36
	Total Scheduled Learning and Teaching Hours:	36
	Hours to be allocated	150
	Allocated Hours	150
Reading List	*The reading list for this module can be accessed via the following link:* https://uwe.rl.talis.com/modules/ufcffl-15-m.html	

Part 5: Contributes Towards
This module contributes towards the following programmes of study: