Module Specification

Applied Numerical Methods [TSI]

Version: 2023-24, v4.0, 10 Aug 2023

Contents

Part 1: Information

Module title: Applied Numerical Methods [TSI]

Module code: UFCF9X-12-2

Level: Level 5

For implementation from: 2023-24

UWE credit rating: 12

ECTS credit rating: 6

College: College of Arts, Technology and Environment

School: CATE School of Computing and Creative Technologies

Partner institutions: None

Field: Computer Science and Creative Technologies

Module type: Module

Pre-requisites: None

Excluded combinations: None

Co-requisites: None

Continuing professional development: No

Professional, statutory or regulatory body requirements: None

Part 2: Description

Overview: Not applicable

Features: Not applicable

Educational aims: The aim of this module to acquaint the students of most common numerical methods, it’s positive and negative properties and accuracy, to discuss and make interpretation of results, to familiarize the students with typical mathematical tasks solving by standard numerical methods in computer applications and researches. The module adopts the use of modern applied mathematics and engineering calculus software such as Matlab to complete the practical assignment. The module provides a strong practical element giving ample opportunity to learn and practise new skills in applied software design.

Outline syllabus: •Sources and main reasons of errors in computer calculations;
•Conditionally of tasks, stability and convergence of methods;
•Direct and iterative methods of the decision of systems of linear equations;
•Local and global interpolation. Approximation;
•Application of methods of numerical differentiation and integration;
•Iterative methods of the decision of non-linear equations and their systems;
•One-dimensional and N-dimensional optimisation. Search methods;
•Numerical methods of the decision of differential equations and their systems;
•Numerical methods of the decision of equations with partial derivatives;
•Recursive and morphological methods. Cellular automata.

Part 3: Teaching and learning methods

Teaching and learning methods: Learning and teaching will be provided to students in three forms: lectures, practical (laboratory) classes, and independent homeworks. During lectures, theoretical aspects of the course will be provided to students by the teaching staff. Lectures will be supported by lectures video records published and available to the students on e.tsi.lv under the module section. Also, additional materials, like presentations, publications on the internet etc will be presented in e.tsi.lv.

Modern programming software such as Visual Studio C++ will be used in laboratory classes (students’ choice) for the algorithms design in comparison with the results in Matlab from practical classes. In addition to learning activities during taught sessions, students are expected to spend time outside of class on independent learning activities. These include completing individual assignment task (homework), independent reading, practising new skills on personal projects and watching informative videos etc.

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

MO1 Utilise established mathematical techniques and algorithms to solve numerical problems accurately and efficiently.

MO2 Utilise commonly used engineering calculation software (such as MATLAB), leveraging its built-in functions and capabilities, to generate and present real-time results that updates dynamically.

MO3 Perform rigorous testing and analysis of numerical methods, identify error sources and reasons, and apply appropriate error mitigation techniques.

Hours to be allocated: 120

Contact hours:

Independent study/self-guided study = 96 hours

Face-to-face learning = 64 hours

Total = 160

Reading list: The reading list for this module can be accessed at readinglists.uwe.ac.uk via the following link https://rl.talis.com/3/uwe/lists/8DDEBBCE-2EA8-37A4-939D-34CEDF14878C.html?lang=en-gb&login=1

Part 4: Assessment

Assessment strategy: To assess learning outcomes within this course, several types of activities are will be used :
1) In-class test to access students to ability to solve numerical problems.
2)Practical skills assessment -students will use the software to produce solutions to mathematical problems , which they will test and analyse to identify errors, including reasons and mitigations.

Assessment tasks:

In-class test (First Sit)

Description: Computer based test (2 hours)

Weighting: 50 %

Final assessment: Yes

Group work: No

Learning outcomes tested: MO1

Practical Skills Assessment (First Sit)

Description: A practical task where students will be required to produce a dynamic solutions sheet (in software) which can be used to solve complex problems.

Weighting: 50 %

Final assessment: No

Group work: No

Learning outcomes tested: MO2, MO3

Practical Skills Assessment (Resit)

Description: A practical task where students will be required to produce a dynamic solutions sheet (in software) which can be used to solve complex problems.

Weighting: 50 %

Final assessment: No

Group work: No

Learning outcomes tested: MO2, MO3

In-class test (Resit)

Description: computer based test (2 hours)

Weighting: 50 %

Final assessment: Yes

Group work: No

Learning outcomes tested: MO1

Part 5: Contributes towards

This module contributes towards the following programmes of study:

Computer Science and Software Development {Double Degree} [Feb][FT][TSI][4yrs] BSc (Hons) 2021-22

Computer Science and Software Development {Double Degree} [Oct][FT][TSI][4yrs] BSc (Hons) 2021-22

Computer Science and Software Development {Double Degree} [Oct][PT][TSI][5yrs] BSc (Hons) 2020-21

Computer Science and Software Development {Double Degree} [Feb][PT][TSI][5yrs] BSc (Hons) 2020-21