Module Specification

Programming (Course Project) [TSI]

Version: 2023-24, v2.0, 17 Mar 2023

Contents

Part 1: Information

Module title: Programming (Course Project) [TSI]

Module code: UFCFJW-6-0

Level: Level 3

For implementation from: 2023-24

UWE credit rating: 6

ECTS credit rating: 3

Faculty: Faculty of Environment & Technology

Department: FET Dept of Computer Sci & Creative Tech

Partner institutions: Transport and Telecommunication Institute

Delivery locations: Not in use for Modules

Field:

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 the module to teach students how to program and to let them develop the necessary skills of software development, using imperative approach and high-level programming languages, such as C and C++.

Outline syllabus: Software lifecycle. Software development stages (project assignment, work stages developing the project software, task analysis);
User interface design basics (implementing a simple user interface);
Program constructs usage in practice. Modular programs (detailed design of software through task decomposition and algorithm development);
Good coding style (implementing software using proper coding conventions);
Testing and debugging;
Documentation for software products (writing a project report paper);

Part 3: Teaching and learning methods

Teaching and learning methods: 6 hours of lectures are provided to students to explain assign individual assignment, explain requirements and demonstrate past course paper and answer questions about assignment. Rest of time students are completing a course paper. Course paper is delivered as report which has programme code realised by students and description of the developed software.

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

MO1 Understand program life cycle and program development stages

MO2 Apply user interface development basics

MO3 Apply modular programming

MO4 Apply proper coding style

MO5 Apply basics of documenting software products

MO6 Complete decomposition of programming tasks into smaller logical parts and creation of algorithms that implement said subtasks

MO7 Implement algorithms using high level programming languages

MO8 Use of C++ operators

MO9 Define and use basic data structures

MO10 Implement basic data processing methods

MO11 Use of development environments and debuggers for program creation and testing purposes

MO12 Develop of simple console applications with basic user interfaces

MO13 Choosing and applying existing data processing algorithms

MO14 Usage of development environment (as example Visual Studio)

Hours to be allocated: 60

Contact hours:

Independent study/self-guided study = 48 hours

Face-to-face learning = 32 hours

Total = 80

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/565B2E25-6B07-C0C8-483D-7D527B92F01F.html?lang=en-gb&login=1

Part 4: Assessment

Assessment strategy: This module assessment consists of one element, which is course paper delivered in form of report. The course paper is delivered in electronic form using TSI LMS and checked by the teacher.

Assessment components:

Written Assignment (First Sit)
Description: Course Paper
Weighting: 100 %
Final assessment: Yes
Group work: No
Learning outcomes tested: MO1, MO10, MO11, MO12, MO13, MO14, MO2, MO3, MO4, MO5, MO6, MO7, MO8, MO9

Written Assignment (Resit)

Description: Course Paper
Weighting: 100 %
Final assessment: No
Group work: No
Learning outcomes tested: MO1, MO10, MO11, MO12, MO13, MO14, MO2, MO3, MO4, MO5, MO6, MO7, MO8, MO9

Part 5: Contributes towards

This module contributes towards the following programmes of study: