



Module Specification

Logical Programming [TSI]

Version: 2023-24, v1.0, 04 Mar 2021

Contents

Module Specification	1
Part 1: Information	2
Part 2: Description	2
Part 3: Teaching and learning methods	3
Part 4: Assessment.....	4
Part 5: Contributes towards	6

Part 1: Information

Module title: Logical Programming [TSI]

Module code: UFCFEX-6-2

Level: Level 5

For implementation from: 2023-24

UWE credit rating: 6

ECTS credit rating: 3

College: College of Arts, Technology and Environment

School: CATE School of Computing and Creative Technologies

Partner institutions: Transport and Telecommunication Institute

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 is to give an understanding of logic programming principles, using PROLOG programming language as an example.

Outline syllabus: Introduction to logic programming paradigm;
Predicate logic;

Introduction to PROLOG programming language. Elements, syntax;
Relations, rules, structures;
Declarative and procedural meaning;
Automaton representation in PROLOG language;
Lists and structure trees;
Route planner;
Manipulating trees.

Part 3: Teaching and learning methods

Teaching and learning methods: Learning and teaching will be provided to students in two forms: lectures and labs. During lectures, theoretical aspects of the course will be provided to students by the teaching staff. Lectures will be supported by presentation published and available to the students on e.tsi.lv under the module section. Also, additional materials, like code examples, text books, publications on the internet, official documentation, videos etc will be presented in e.tsi.lv. During labs, each student receives an individual task to perform. PROLOG is considered as logical programming language. In addition to learning activities during taught sessions, students are expected to spend time outside of class on independent learning activities. These might include completing assignment tasks, independent reading, practising new skills on personal projects.

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

MO1 Understand and use the basics of logical programming

MO2 Understand and apply the underlying mathematics

MO3 Understanding predicate expressions

MO4 Read PROLOG programs created by others, using those as examples to solve problems.

MO5 Develop simple programs in PROLOG language

MO6 Dissecting the subject area into phrases, then translating phrases into predicate expressions

MO7 Widening personal knowledge in logical programming, relying on experience from this course

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](https://rl.talis.com/3/uwe/lists/9456BB8B-6E7F-95FE-2929-D240AFC212F6.html?lang=en-gb&login=1) via the following link <https://rl.talis.com/3/uwe/lists/9456BB8B-6E7F-95FE-2929-D240AFC212F6.html?lang=en-gb&login=1>

Part 4: Assessment

Assessment strategy: This module assessment is split into two components (A – Exam, B – Labs):

A1 - final 2-hour examination which will assess the students understanding of taught material that forms part of the learning outcomes but cannot easily be assessed through practical tasks.

B1 – series of labs, exploring basic principles of logical programming using PROLOG programming language as example. An application and its source code should be provided to the teaching staff in form of report.

B2 – series of in-class tests with theoretical and practical tasks.

Assessment tasks:

Examination - Component A (First Sit)

Description: Examination

Weighting: 40 %

Final assessment: Yes

Group work: No

Learning outcomes tested: MO1, MO2, MO3, MO7

Portfolio - Component B (First Sit)

Description: A series of labs, exploring basic principles of logical programming using PROLOG programming language as example. An application and its source code should be provided to the teaching staff in form of report.

Weighting: 36 %

Final assessment: No

Group work: No

Learning outcomes tested: MO1, MO3, MO4

In-class test - Component B (First Sit)

Description: A series of in-class tests with theoretical and practical tasks.

Weighting: 24 %

Final assessment: No

Group work: No

Learning outcomes tested: MO1, MO2, MO3, MO5, MO6

In-class test - Component B (Resit)

Description: series of 3 quizzes with theoretical and practical tasks. Answers for quizzes should be provided to the teaching staff through e.tsi.lv system in the module section

Weighting: 24 %

Final assessment: No

Group work: No

Learning outcomes tested: MO1, MO2, MO3, MO5, MO6

Examination - Component A (Resit)

Description: Examination

Weighting: 40 %

Final assessment: No

Group work: No

Learning outcomes tested: MO1, MO2, MO3, MO7

Portfolio - Component B (Resit)

Description: series of 5 labs, exploring basic principles of logical programming using PROLOG programming language as example. An application and its source code should be provided to the teaching staff in form of report.

Weighting: 36 %

Final assessment: No

Group work: No

Learning outcomes tested: MO1, MO3, MO4

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