# MODULE SPECIFICATION

| Part 1: Information | | | |
|---|---|---|---|
| Module Title | Functional Programming [TSI] | | |
| Module Code | UFCFXX-6-2 | Level | Level 5 |
| For implementation from | 2022-23 | | |
| UWE Credit Rating | 6 | ECTS Credit Rating | 3 |
| 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 | | |
| PSRB Requirements | None | | |

## Part 2: Description

**Educational Aims:** The aim of this module is to give a basic understanding of functional programming principles using Haskell programming language as an example.

**Outline Syllabus:** Introduction to functional programming paradigm;
Introduction to Haskell language;
Functional language elements and data types (based on HASKELL example);
Recursion. Program execution;
Lambda-calculus;
Imperative program analogies

**Teaching and Learning Methods:** Learning and teaching will be provided to students in forms of lectures, labs, practical classes. 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.

## Part 3: Assessment

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. This component represents 50% of final module mark.
The practical assignment component should be completed individually (i.e. this is not group work) and represents 50% of final module grade.
B1 – series of practical tasks (labs), exploring basic principles of functional programming using HASKELL programming language. An application and its source code should be provided to the teaching staff in form of report.

| First Sit Components | **Final Assessment** | **Element weighting** | **Description** |
|---|---|---|---|
| Examination - Component A | ✓ | 50 % | Examination |
| Portfolio - Component B | | 50 % | A series of practical tasks (labs), exploring basic principles of functional programming using HASKELL programming language An application and its source code should be provided to the teaching staff in form of report |
| Resit Components | **Final Assessment** | **Element weighting** | **Description** |
| Examination - Component A | | 50 % | Examination |
| Portfolio - Component B | | 50 % | A series of practical tasks (labs), exploring basic principles of functional programming using HASKELL programming language. An application and its source code should be provided to the teaching staff in form of report. |

## 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 and use the basics of functional programming | MO1 |
| Understand and use the basics of lambda-calculus | MO2 |
| Read HASKELL programs created by others, using those as examples to solve problems | MO3 |
| Dissect the subject area into functions, using functions as abstractions | MO4 |
| Apply functional programming to solve practical problems | MO5 |
| Develop simple programs in Haskell language | MO6 |
| Widening personal knowledge in functional programming, relying on experience from this course | MO7 |

Contact Hours

| **Independent Study Hours:** | |
|---|---|
| Independent study/self-guided study | 48 |

| | | |
|---|---|---|
| | **Total Independent Study Hours:** | 48 |
| | **Scheduled Learning and Teaching Hours:** | |
| | Face-to-face learning | 32 |
| | **Total Scheduled Learning and Teaching Hours:** | 32 |
| | **Hours to be allocated** | 60 |
| | **Allocated Hours** | 80 |
| Reading List | *The reading list for this module can be accessed via the following link:*<br>https://rl.talis.com/3/uwe/lists/1E04FAF9-1493-FE10-A739-4E142FB30146.html?lang=en-gb&login=1 | |

## Part 5: Contributes Towards

This module contributes towards the following programmes of study:

Computer Science and Software Development [Oct][FT][TSI][4yrs] BSc (Hons) 2020-21

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

Computer Science and Software Development [Feb][FT][TSI][4yrs] BSc (Hons) 2020-21

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