# MODULE SPECIFICATION

| Part 1: Information | | | |
|---|---|---|---|
| Module Title | Programming Languages Concepts [TSI] | | |
| Module Code | UFCFLW-12-0 | Level | Level 3 |
| For implementation from | 2020-21 | | |
| UWE Credit Rating | 12 | ECTS Credit Rating | 6 |
| 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 main aim of the module is to provide a systematic study of the principles of design, evaluation and use of modern programming languages. The module gives an overview of PL design principles and discusses theoretical issues of modern languages semantic and operating principles.

**Outline Syllabus:** Defining and problems of programming languages;
Types of Programming Languages;
The expressions in programming languages;
The actions and statements in the programs;
The syntax and grammars;
Parsing and derivations;
The grammar for expressions;
Versions of grammars;
Semantics of Programming Languages;
Objects and data types;
Scalar and array types;
Records and sets;
Pointers;

Variable attributes;
Data type conversions;
Data type conversions;
Introduction to the subroutines;
Parameter-passing methods;
Scopes and activations;
Exception handling;
Abstract data types;
Object-oriented programming;

**Teaching and Learning Methods:** Learning and teaching will be provided to students in forms of lectures, practical classes and laboratory works. Lectures are supported by presentations published and available to the students on e.tsi.lv under the module section. Also, additional materials, like publications on the internet, videos etc will be presented in e.tsi.lv.
Practical classes are devoted to solving and discussion of practical tasks, regarding grammar, scopes and activations, use of subroutines, parameter passing methods. Later, students' knowledge and understanding will be tested using control work (in written form).

**Part 3: Assessment**

This module assessment is split into two components (A – Exam, B – Practical Assignments):
The practical assignment component should be completed individually (i.e. this is not group work) and represents 50% of final module mark.
A1: - final 2-hour closed books examination which covers theoretical aspects of the module
The practical assignments and labs has following elements
B2: - a sequence of in-class tests about module topics
B1: - a sequence of 4 individual assignments. For each assignment a report shall be provided by the student

| First Sit Components | **Final Assessment** | **Element weighting** | **Description** |
|---|---|---|---|
| Examination - Component A | ✓ | 35 % | Examination |
| Portfolio - Component B | | 40 % | 5 sequential laboratory works |
| In-class test - Component B | | 25 % | a series of tests on module topics |
| Resit Components | **Final Assessment** | **Element weighting** | **Description** |
| Examination - Component A | | 35 % | Written exam |
| Portfolio - Component B | | 40 % | 5 sequential laboratory works |
| In-class test - Component B | | 25 % | 4 Written test |

**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** |
|---|---|
| Know and use the basic principles and concepts that define the construction and application of programming language | MO1 |
| Select PL for a specific software project, considering particularities of each programming language type | MO2 |

| | | |
|---|---|---|
| | To learn by themselves new programming languages, considering knowledge about common elements in PLs and PLs construction | MO3 |
| | Consider the cost of the development and application of PL in a particular domain | MO4 |
| | Understand limitations of various high-level languages | MO5 |
| | Understanding the PLs trends | MO6 |
| | Be able to track trends and directions in development of programming languages | 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** | 120 |
| | **Allocated Hours** | 80 |

| Reading List | |
|---|---|
| | *The reading list for this module can be accessed via the following link:*<br><br>https://rl.talis.com/3/uwe/lists/0BB6B0D6-F000-5402-FFE2-100352C726B5.html?lang=en-gb&login=1 |

| **Part 5: Contributes Towards** |
|---|
| This module contributes towards the following programmes of study:<br><br>Computer Science and Software Development [Oct][PT][TSI][5yrs] BSc (Hons) 2020-21 BSc (Hons) 2020-21<br>Computer Science and Software Development [Oct][FT][TSI][4yrs] BSc (Hons) 2020-21<br>Computer Science and Software Development [Feb][FT][TSI][4yrs] BSc (Hons) 2020-21<br>Computer Science and Software Development [Feb][PT][TSI][5yrs] BSc (Hons) 2020-21 BSc (Hons) 2020-21 |