

## Liverpool John Moores University

Title: INTRODUCTION TO PROGRAMMING  
Status: Definitive  
Code: **4500CSQR** (127397)  
Version Start Date: 01-08-2021

Owning School/Faculty: Computer Science and Mathematics  
Teaching School/Faculty: Oryx Universal College WLL

Team	Leader
Denis Reilly	Y

**Academic Level:** FHEQ4  
**Credit Value:** 20  
**Total Delivered Hours:** 66  
**Total Learning Hours:** 200  
**Private Study:** 134

### Delivery Options

Course typically offered: Semester 1

Component	Contact Hours
Lecture	33
Practical	22
Workshop	11

**Grading Basis:** 40 %

### Assessment Details

Category	Short Description	Description	Weighting (%)	Exam Duration
Technology	AS1	Simple Application	40	
Technology	AS2	Complex Application	60	

### Aims

*To gain an understanding of how software is developed.*  
*To become conversant with a range of computer programming paradigms.*  
*To develop problem solving skills in computing.*  
*To prepare students for software development at higher levels, both in work and study.*

## Learning Outcomes

After completing the module the student should be able to:

- 1 Apply knowledge of programming constructs and basic algorithms.
- 2 Demonstrate problem solving skills by producing simple programming solutions.
- 3 Evaluate alternatives and make sound judgements regarding programming solutions.
- 4 Investigate integrated development environments & application programming interfaces.
- 5 Demonstrate basic knowledge of the object oriented programming paradigm.

## Learning Outcomes of Assessments

The assessment item list is assessed via the learning outcomes listed:

Simple Application	1	2		
Complex Application	3	4	5	

## Outline Syllabus

*Programming Overview & History*  
*The Language & IDE*

*Basic Elements*  
*Variables & Constants*  
*Operators, Expressions & Statements*

*Using Objects & Methods*

*I/O & File I/O*

*Selection*  
*Boolean Operators & Expressions*  
*If, If-Else & Switch-Case*

*Iteration*  
*While, For & Do-While*  
*Break & Continue*

*User-Defined Methods*  
*Return Types*  
*Parameters*  
*Scope*

*Arrays*  
*of Value & Reference Type*  
*For-Each*

*Multi-Dimensional*

*Object Oriented Design*  
*Introduction to UML*

*User-Defined Classes*  
*Members*  
*Constructors*

*Exceptions & Event Handling*  
*Try, Catch & Finally*  
*Throwing*

*Graphical User Interfaces*  
*Event Driven Programming*

## **Learning Activities**

Learning activities include lectures and tutorials where students are encouraged to ask questions / discuss scenarios and supported labs where students are encouraged to put theory gained through lectures and tutorials into practice. Directed reading against appropriate industry and research sources further reinforces learning.

## **Notes**

This module imparts upon students, basic programming and problem solving skills that help prepare them for further study in more specialised software development topics.