

## Liverpool John Moores University

Warning: An incomplete or missing proforma may have resulted from system verification processing

Title: Introduction to Programming  
Status: Definitive  
Code: **4000SEQR** (129294)  
Version Start Date: 01-08-2021

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

Team	Leader
Bo Zhou	Y

**Academic Level:** FHEQ4      **Credit Value:** 20      **Total Delivered Hours:** 44  
**Total Learning Hours:** 200      **Private Study:** 156

### Delivery Options

Course typically offered: S1 & S2 & Summer

Component	Contact Hours
Lecture	22
Practical	22

**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 familiar with a range of computer programming paradigms.*  
*To develop basic problem solving skills in computing.*  
*To prepare students for software development at higher levels.*

## 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*  
*Basic I/O & File I/O*  
*Selection*  
*Boolean Operators & Expressions*  
*If, If-Else & Switch-Case*  
*Iteration*  
*While, For & Do-While*  
*Break*  
*Collections*  
*Array and ArrayList*  
*String and char Types*  
*User-Defined Methods*  
*Return Types*  
*Parameters*  
*Scope*  
*User-Defined Classes*  
*Members*  
*Constructors*  
*Exceptions & Event Handling*  
*Try, Catch & Finally*

## Learning Activities

Learning activities include lectures where students are encouraged to ask questions and discuss scenarios, and supported labs where students are encouraged to put theory gained through lectures into practice.

## **Notes**

This module delivers programming and problem solving skills, with no prior assumptions of programming experience. Given the importance of programming to computer science this module will encourage students to study more specialized software development topics.