

Liverpool John Moores University

Title: PROGRAMMING AND SOFTWARE DEVELOPMENT
Status: Definitive
Code: **4544NCCG** (129506)
Version Start Date: 01-08-2021

Owning School/Faculty: Computer Science and Mathematics
Teaching School/Faculty: Nelson Campus

Team	Leader
Silvester Czanner	Y
Robert Askwith	

Academic Level: FHEQ4 **Credit Value:** 20 **Total Delivered Hours:** 60
Total Learning Hours: 200 **Private Study:** 140

Delivery Options

Course typically offered: S1, S2 and NS2 (S2 for Jan)

Component	Contact Hours
Lecture	60

Grading Basis: 40 %

Assessment Details

Category	Short Description	Description	Weighting (%)	Exam Duration
Practice	Exercise	Coding Exercise	67	
Report	Assignment	Assignment	33	

Aims

This module will introduce students to the software development life cycle and to designing, coding, implementing and testing programmes to meet specified briefs.

Learning Outcomes

After completing the module the student should be able to:

- 1 Define basic algorithms to carry out an operation and outline the process of programming an application.
- 2 Explain the characteristics of procedural, object-orientated and event-driven programming, conduct an analysis of a suitable Integrated Development Environment (IDE).
- 3 Implement basic algorithms in code using an IDE
- 4 Determine the debugging process and explain the importance of a coding standard.
- 5 Describe and implement the stages of the software development lifecycle

Learning Outcomes of Assessments

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

Coding Exercise	1	3	
Assignment	2	4	5

Outline Syllabus

Major theme: Programming

- *Algorithm definition: the relationship between algorithms and code. The generation process of code. Roles of the pre-processor, compiler and linker, interpreter.*
- *Characteristics of procedural, object-orientated and event-driven programming. Definitions of: data types, methods, control structures, iteration, scope, parameter passing, classes, inheritance and events.*
- *Implement basic algorithms in code: developing simple applications using a suitable language and IDE.*
- *Debugging processes. Importance of a coding standard. Documentation of the debugging process*
- *Testing and integration & test environments*
- *Importance of feasibility study: Requirement gathering techniques. Feasibility criteria*
- *Carry out software development: Identify requirements: stakeholders; consideration of alternate solutions and security considerations; quality assurance required. Report documentation structure.*
- *Systems analysis terminology and tools: data stores and entities; data flows; process representation techniques, examples relevant to methodology chosen.*
- *Techniques: Flowcharts; Pseudocode; Formal specification Methods; Event/State/Data Driven; Finite State Machines (extended-FSM)/FSP; problem of e-FSM state explosion; reachability analysis, safety, liveness properties; Automatic analysis and animation tools.*

Minor theme: Software development processes

Software development lifecycles: Lifecycle models, both predictive and adaptive. Lifecycle stages: feasibility study, analysis, design, implementation, testing, review or analysis, design, implementation, maintenance, planning; requirements traceability.

Learning Activities

Lectures

These will not normally be traditional didactic lectures in which the student plays little active part, but will be delivered in small groups of up to 20

Notes

-