

Course descriptor F21AS

Course code	F21AS
Course title	Advanced Software Engineering
Credits	15
School	Mathematics and Computer Sciences
SCQF Level	11
Semester	2
Aims	<p>This course aims to:</p> <ul style="list-style-type: none"> • To consolidate proficiency in imperative programming and software development • To further develop object oriented programming and object oriented design methods • To provide knowledge of simple data structures and algorithms • To introduce concurrent programming techniques • To instil understanding of the concepts and benefits of advanced software engineering methods • To give further practical experience of the use of UML in software engineering • To give practical experience of developing a substantial software engineering team project • To enable the deployment of patterns in software engineering
Syllabus	<ul style="list-style-type: none"> • Data structures: stacks, queues, lists, priority queues, binary trees • Algorithms: searching (linear and binary) and sorting • Advanced object oriented design techniques • Thread based programming: thread creation and interaction, shared variables and synchronisation • Methodologies in software engineering practice; Unified Modelling Language; design patterns • Project planning and management in software engineering • Comparison of agile and plan driven approaches

Learning Outcomes	
Subject Mastery	<ul style="list-style-type: none"> • Skill in the use of UML notation and translation of UML designs to working programs • Understanding of basic data structures and algorithms and ability to critically evaluate their appropriateness and limitations for a range of moderately complex problems. • Demonstration of skill in design and implementation of practical GUI based and threaded applications • To demonstrate a critical understanding of modern software engineering practice and be able to evaluate the strengths

	<p>and weaknesses of current software engineering methods and techniques</p> <ul style="list-style-type: none"> • To be able to choose appropriate metrics to measure software quality and quantity in a modern software engineering environment • To be able to choose a suitable software development environment and development methodology for specific software development tasks and justify the choice
Personal Abilities	<ul style="list-style-type: none"> • Appreciation of use of methodology to ground system analysis, design and development • Understanding of different programming paradigms and their inter-relation • Practice in working in a group, choosing a methodology, reaching a consensus, and working with others to a deadline • Taking responsibility for own work, taking responsibility in the development of resources, critical reflection on development process and work undertaken by self. • Effective appreciation of professional standards in modern software engineering practice. • Showing initiative, creativity and team working skills in collaborative software development •

Assessment method	50% written examination, 50% coursework
-------------------	---