ROBERT GORDON UNIVERSITY ABERDEEN

MODULE DESCRIPTOR

Module Title

Advanced Programming

Reference	CM2710	Version	3
Created	January 2023	SCQF Level	SCQF 8
Approved	May 2019	SCQF Points	30
Amended	June 2023	ECTS Points	15

Aims of Module

To provide students with an introduction to the principles of advanced programming concepts and techniques including the modelling, design, implementation and testing of such programs.

Learning Outcomes for Module

On completion of this module, students are expected to be able to:

1	Demonstrate an understanding of object oriented modelling and design concepts.
2	Compare between alternative designs of algorithms and data structures.
3	Use a range of industry standard data formats to store and transfer data.
4	Discuss the properties of good software design including the nature and the role of associated documentation.
5	Analyse software quality assurance processes involving industry standard techniques.

Indicative Module Content

Object-oriented design and programming, design patterns, sorting and searching, recursive algorithms, computational complexity. Implementation of standard data abstractions using: arrays, lists, trees, hash tables. File handling, file formats (including CSV), data exchange formats (including JSON, XML), RSS feeds, web APIs. Exceptions, separation of concerns, single responsibility principle, code comments, self-documenting code, code standards and best practice guides, UML diagrams, requirements documentation, test strategies. Levels of verification (analysis, demonstration, test, formal proof, inspection etc.) and testing (unit, integration, systems, and acceptance), test coverage, automation of tests, unit testing frameworks, acceptance criteria, business implications.

Module Delivery

The module is delivered in Blended Learning mode using structured online learning materials/activities and directed study, facilitated by regular online tutor support. Workplace Mentor support and work-based learning activities will allow students to contextualise this learning to their own workplace. Face-to-face engagement occurs through annual induction sessions, employer work-site visits, and modular on-campus workshops.

Module Ref:	CM2710 v3

Indicative Student Workload	Full Time	Part Time
Contact Hours	30	N/A
Non-Contact Hours	30	N/A
Placement/Work-Based Learning Experience [Notional] Hours	240	N/A
TOTAL	300	N/A
Actual Placement hours for professional, statutory or regulatory body	240	

ASSESSMENT PLAN

If a major/minor model is used and box is ticked, % weightings below are indicative only.

Component 1

Type:	Coursework	Weighting:	100%	Outcomes Assessed:	1, 2, 3, 4, 5
Description:	This coursework will consist of a software design and development exercise and a discussion on aspects of software documentation and software quality assurance within the workplace.				

MODULE PERFORMANCE DESCRIPTOR

Explanatory Text

The calculation of the overall grade for this module is based on 100% weighting of C1. An overall minimum grade D is required to pass the module.

Module Grade	Minimum Requirements to achieve Module Grade:
A	The student must achieve an A in C1.
B	The student must achieve a B in C1.
C	The student must achieve a C in C1.
D	The student must achieve a D in C1.
E	The student must achieve an E in C1.
F	The student must achieve an F in C1.
NS	Non-submission of work by published deadline or non-attendance for examination

Module Requirements

Prerequisites for Module	CM1705 Fundamentals of Programming, or equivalent.
Corequisites for module	None.
Precluded Modules	None.

INDICATIVE BIBLIOGRAPHY

1. LUTZ, M., 2013. Learning Python. 5th Ed. Beijing, China: O?Reilly.
2. PADMANBHAN T.R., 2016. Programming with Python. Singapore: Springer.
3. BUELTA, J., 2020. Python Automation Cookbook: 75 Python Automation Ideas for Web Scraping, Data Wrangling, and Processing Excel, Reports, Emails, and More, 2nd Ed. Birmingham, UK: Packt Publishing.
4. PHILLIPS, D., 2015. Python 3 object-oriented programming unleash the power of Python 3 objects. 2nd ed. Birmingham, UK: Packt Publishing.
5. SOMMERVILLE, I., 2015. Software Engineering. 10th ed. Harlow: Pearson