ROBERT GORDON UNIVERSITY ABERDEEN

This Version is No Longer Current

The latest version of this module is available here

MODULE DESCRIPTOR

Module Title

Object-oriented Programming

Reference	CMM024	Version	2
Created	April 2017	SCQF Level	SCQF 11
Approved	July 2016	SCQF Points	15
Amended	August 2017	ECTS Points	7.5

Aims of Module

To introduce and explore the key concepts of object-oriented programming. To develop the student's skill in the practical design, development and testing of simple object oriented applications. To extend the student's knowledge and proficiency in object oriented design to include class design by inheritance. To introduce issues of object oriented design that arise in the development of interactive programs that incorporate a graphical user interface.

Learning Outcomes for Module

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

1	Analyse a set of system requirements and design, develop and test a solution which satisfies those requirements using an object oriented programming language.
2	Interpret and describe class behaviour and inter-class relationships in an object-oriented design using appropriate notation.
3	Demonstrate knowledge and understanding of object-oriented design concepts including class composition, association, and inheritance.
4	Implement interactive object-oriented programs that incorporate a graphical user interface and hierarchies of classes, using an event handling model to handle user interaction with components.

Indicative Module Content

The module will introduce and then develop student's knowledge and skills involving the basic concepts of object oriented programming. The module will then progress in the use of further object oriented techniques (especially inheritance) in the design of class hierarchies. The concept of inheritance will explored within the context of the design and implementation of interactive object oriented programs that incorporate a graphical user interface. The module content will introduce and use the Unified Modelling Language (UML) to guide and document the design process and describe class behaviours and interactions. Module content will cover: Key concepts of basic programming structures: variables, decision controls, repetition controls, arrays. Key concepts of object oriented programming including: classes, instance objects, instance members, methods, class objects, class members and class methods. Hierarchical mechanisms for object re-use through composition, association, inheritance, interfaces, abstract classes, and polymorphism. Creation of simple GUIs and incorporating an appropriate event handling model and exception handling, making use of graphical toolkits.

Module Delivery

Key concepts are introduced and illustrated in lectures. Tutorials are used to develop and evaluate design ideas (using Unified Modelling Language) before implementation. The laboratory sessions are used to progress through a series of exercises intended to develop understanding of the lecture content and to develop proficiency in the practical application of object oriented programming skills, including the use of existing packages for development of graphical user interfaces and programming tools that aid the development process.

Indicative Student Workload

	Full Time	Part Time
Contact Hours	66	66
Non-Contact Hours	84	84
Placement/Work-Based Learning Experience [Notional] Hours	N/A	N/A
TOTAL	150	150
Actual Placement hours for professional, statutory or regulatory body		

ASSESSMENT PLAN

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

Component 1

Type:	Practical Exam	Weighting:	100%	Outcomes Assessed:	1, 2, 3, 4
Description:	Multi part Computer-based programming assessment				

MODULE PERFORMANCE DESCRIPTOR

Explanatory Text

The calculation of the overall grade for this module is based on combined marks from all parts from a multi part practical programming assessment.

Module Grade	Minimum Requirements to achieve Module Grade:
A	A grade in Component 1
B	B grade in Component 1
C	C grade in Component 1
D	D grade in Component 1
E	E grade in Component 1
F	F grade in Component 1
NS	Non-submission of work by published deadline or non-attendance for examination

Module Requirements

Prerequisites for Module	None.
Corequisites for module	None.
Precluded Modules	None.

INDICATIVE BIBLIOGRAPHY

1	DEITEL, P.J. AND DEITEL, H., 2015. Java: Early Objects 10th ed. Prentice Hall.
2	SCHILDT,H.,2019. Java: The Complete Reference. 11th ed. McGraw Hill.
3	LIANG, Y, D., 2019. Introduction to Java Programming and Data Structures. Comprehensive version. Pearson.
4	HORSTMANN, C., 2016. Core Java. 10th ed. Prentice Hall.
5	HORSTMANN, C, 2016. Big Java: Late Objects. 2nd ed. Wiley.