Anglia Ruskin University

Module Definition Form (MDF)

Module code: MOD005244	**Version:** 1 **Date Amended:** 14/Mar/2016

1. Module Title
Introduction to Game Programming

2a. Module Leader
Senir Dinar

2b. Department
Department of Computing and Technology

2c. Faculty
Faculty of Science and Technology

3a. Level
4

3b. Module Type
Standard (fine graded)

4a. Credits
30

4b. Study Hours
300

5. Restrictions

Type	**Module Code**	**Module Name**	**Condition**
Pre-requisites:	None		
Co-requisites:	None		
Exclusions:	None		
Courses to which this module is restricted:	BSc (Hons) Computer Gaming Technology		

LEARNING, TEACHING AND ASSESSMENT INFORMATION

6a. Module Description

This module provides an introduction to the high level programming knowledge a game programmer needs (although it requires no prior programming experience). The student will use industry-standard tools and techniques to design, implement, test and document simple programs using a current programming language.

The module will enable students to understand the principal components of high-level programming code, laying the foundation for subsequent game development modules which require structured programming ability. It will emphasise the principles of good programming practice and introduce the techniques required to develop games which:

* are robust and efficient

* implement the required game mechanics

* contain modular and reusable code

* consists of elegant, easy to read code

Summative assessment will address the student's knowledge of programming theory, syntax and best practice. The module is assessed by a coursework assignment which will test the student's application of that knowledge by presenting a game mechanic problem and task requirements, to be solved by the design, implementation and testing of game objects, plus associated documentation. Formative exercises will be set at intervals through the module for peer review and feedback. By the end of the module, students should have sufficient mastery of a high-level programming language to allow them to design, implement and test simple game mechanics.

The skills taught within the module are intended to be directly transferable to the games industry and will provide a suitable foundation for students who will be expected to apply programming skills in future modules within the BSc (Hons) Computer Gaming Technology degree. These skills are readily transferable outside of the games industry and will provide a strong foundation for other software development disciplines.

6b. Outline Content

An introduction to:

* Control structures: sequence, selection, iteration and Boolean algebra.

* Variables, constants, data types and operators.

* Program design, program structure, testing/debugging methodologies, documentation.

* Functions/procedures, algorithms (such as searching and sorting).

* Simple data structures, such as arrays and records.

* File handling.

* The object oriented paradigm: information hiding, encapsulation, classes and methods.

* Principals of good programming practice such as reusability, maintainability and intellectual property considerations using examples appropriate to the game development industry.

6c. Key Texts/Literature

The reading list to support this module is available at: http://readinglists.anglia.ac.uk/modules/mod005244

6d. Specialist Learning Resources
Students will need regular and frequent access to both hardware and software in order to develop their programming skills both on and off campus. A suitable programming environment and associated e-learning materials must be readily available. Open-access labs equipped with appropriate software and Internet access are required on the delivery campus. Students may download the programming environments and tutorial material used in this module for use at home.

7. Learning Outcomes (threshold standards)

No.	Type	On successful completion of this module the student will be expected to be able to:
1	Knowledge and Understanding	Understand and describe the fundamental structures and syntax of programming in a popular high-level programming language.
2	Knowledge and Understanding	Understand and describe the fundamental concepts of Object Oriented Programming in a popular high-level programming language.
3	Knowledge and Understanding	Select and use appropriate techniques and tools to design solutions to a range of simple problems which can be coded in a high-level programming language.
4	Intellectual, practical, affective and transferrable skills	Create programs using appropriate syntax and structures in a high- level programming language for a range of simple game related problems.
5	Intellectual, practical, affective and transferrable skills	Test a range of simple programs by selecting and using appropriate techniques.
6	Intellectual, practical, affective and transferrable skills	Create code which adheres to given guidelines of good programming practice suitable for the language used.

8a. Module Occurrence to which this MDF Refers

Year	Occurrence	Period	Location	Mode of Delivery
2017/8	ZZF	Template For Face To Face Learning Delivery		Face to Face

8b. Learning Activities for the above Module Occurrence			
Learning Activities	**Hours**	**Learning Outcomes**	**Details of Duration, frequency and other comments**
Lectures	24	1-6	2 x 1 hour lectures each week
Other teacher managed learning	24	1-6	2 hour lab each week
Student managed learning	252	1-6	On-line course materials supporting the lecture and tutorial series are to be provided via VLE
TOTAL:	300		

9. Assessment for the above Module Occurrence					
Assessment No.	**Assessment Method**	**Learning Outcomes**	**Weighting (%)**	**Fine Grade or Pass/Fail**	**Qualifying Mark (%)**
010	Coursework	1-6	100 (%)	Fine Grade	30 (%)
A coursework assignment which will test the student's application of programming knowledge by presenting a problem and task requirements, to be solved by the design, implementation and testing of a computer program, including associated documentation and in class verification (equivalent to 5000 words).					

In order to pass this module, students are required to achieve an overall mark of 40%.
In addition, students are required to:
(a) achieve the qualifying mark for each element of fine graded assessment of as specified above
(b) pass any pass/fail elements