

Module Information

Module Identifier	CS23710
Module Title	C and Unix Programming
Academic Year	2015/2016
Co-ordinator	<u>Mr David Ernest Price (mailto:dap@aber.ac.uk?subject=CS23710)</u>
Semester	Semester 1
Co-Requisite	<u>CC21120 (?m=CC21120)</u> or <u>CS21120 (?m=CS21120)</u>
Pre-Requisite	<u>CS21120 (?m=CS21120)</u>
Pre-Requisite	<u>CS12320 (?m=CS12320)</u>
Pre-Requisite	<u>CS21120 (?m=CS21120)</u>
Other Staff	<u>Mr David Ernest Price (mailto:dap@aber.ac.uk?subject=CS23710)</u> <u>Dr Frederick William Long</u> <u>(mailto:fwl@aber.ac.uk?subject=CS23710)</u>

Course Delivery

Delivery Type	Delivery length / details
Lecture	22 lectures
Practical	11 x 2hrs

Assessment

Assessment Type	Assessment length / details	Proportion
Semester Assessment	Worksheets	10%
Semester Assessment	1 piece of software development coursework (Approx 32 hours)	50%
Semester Exam	1.5 Hours written exam	40%
Supplementary Exam	1.5 Hours supplementary exam Resit failed examination and/or resubmission of failed/non-submitted coursework components or ones of equivalent value	100%

Learning Outcomes

On successful completion of this module students should be able to:

1. Have consolidated and extended their knowledge of programming languages.
2. be able to utilize the new skills acquired to analyse and solve problems using the C language.

3. be able to exercise significant judgment in being able to evaluate whether C or Java would be the most appropriate language to use for a range of situations.
4. be able to use specialized skills, namely in the use of operating system facilities.
5. have reviewed the importance of standards by means of studying the specifications of ANSI standard C and the POSIX operating systems interface.

Brief description

This module aims to teach the basic features of the C programming language, the structure of the UNIX operating system and the UNIX programming interface. Many of the facilities of C will be introduced by comparison with those of Java, the emphasis being on the rationale behind the language design decisions for the two programming languages. Emphasis will also be placed on C programming incorporating good modular design, readability, portability, and use of appropriate development tools.

The latter part of the course will address an introduction to the structure of the UNIX operating system and the interface presented by functions and system calls for use from C.

The module will also consider the relevance of standards, both in C and in UNIX.

Content

1. Introduction - 1 lecture

An introduction to the module.

2. Basic Concepts of C - 2 Lectures

History of the C language, philosophical differences between C language design and Java. ANSI C. Basic form of a C program compared with that of a Java program. Using the compiler.

3. Control Structures - 2 Lectures

Sequence, branching and iteration in C compared with that of Java.

4. Basic Data Structures - 1 Lecture

Review of basic data types and operators in C.

5. Functions - 1 Lecture

Discussion of ways in which functions are implemented, and used in C, including parameter passing mechanisms.

6. Composite Data Structures - 1 Lecture

A first discussion of Arrays in C. Input/Output.

7. Software Support Tools - 1 Lecture

Make, Lint, Debuggers. Libraries and library utilities. Static/Dynamic analysers.

8. C Programming Style and Portability - 1 Lecture

Language standards. Portability. Programming standards.

9. Arrays, Pointers and Functions - 2 Lectures

A discussion of pointer data types, how they relate to arrays, and how they contrast with Java objects.

10. Dynamic Data Structures - 1 Lecture

Implementation of various record structures and dynamic structures. Pointers. Malloc. Examples in C. Parallels will be drawn with how the internals of Java do this for you.

11. Pitfalls - 1 Lecture

Major problem areas. Design rationale of C and of Java in problem areas.

12. Further Features - 1 Lecture

C preprocessor, header files, conditional inclusion, macro substitution, bitwise operators, casts, enumeration, scope, static and external declarations, separate compilation.

13. Case Studies ? 4 Lectures

A small group of typical ANSI C programs will be presented and discussed as examples of good practice.

14. Exercises ? about 11 Practicals

Formative practicals covering the main features of the C language.

15. The Structure of UNIX - 1 Lecture

An overview of the structure of the UNIX operating system and a description of the major components, including: the kernel, device drivers, IEEE POSIX and the terminal. An introduction to the programming interface to UNIX.

16. The Filesystem and Files - 1 Lecture

The structure of the filesystem and associated operations. The types of file and the operations on files. The function and system calls available from C to interact with files and filesystems.

17. Summary and Conclusions - 1 lecture

Drawing together the topics covered in the module.

Module Skills

Skills Type	Skills details
Application of Number	No.
Communication	Written skills will be needed to complete supporting documents to accompany assessed coursework.

Skills Type	Skills details
Improving own Learning and Performance	See 2 above
Information Technology	The whole module concerns this area.
Personal Development and Career planning	Careful time management will be needed as so to enable students to complete coursework etc.
Problem solving	This is inherent in both the formative practical work and the assessed coursework.
Research skills	The students will need to search for and use relevant technical information while completing practical and assessed coursework.
Subject Specific Skills	Students will develop a detailed knowledge of the internal functioning of a computer system and how programs gain access to the facilities of an operating system.
Team work	No.

Notes

This module is at CQFW (<http://wales.gov.uk/topics/educationandskills/qualificationsinwales/creditqualificationsframework/?lang=en>) Level 5