CS118-15 Programming for Computer Scientists

24/25

Department
Computer Science
Level
Undergraduate Level 1
Module leader
James Archbold
Credit value
15
Module duration
10 weeks
Assessment
Multiple
Study location
University of Warwick main campus, Coventry

Description

Introductory description

This module is a first course in computer programming.

Module aims

The principal aim of this module is to introduce students to problem solving and structured and object oriented programming.

Outline syllabus

This is an indicative module outline only to give an indication of the sort of topics that may be covered. Actual sessions held may differ.

Fundamentals of programming

Introduction to programming: Programming paradigms; From Specification through Algorithms to Implementation; Program compilation and testing.
Building Elements: Preconditions and postconditions; Basic data types; Variables, identifiers and scope.

Control structures: Conditionals; Case statements and loops; Correctness issues when programming with loops.
Methods: Comparison between iteration and recursion.
Object oriented programming.
An introduction to multi-threading.

Programming with objects and classes: Complex data types; Parameter passing by reference and by value; Encapsulation.
Arrays and strings
Class inheritance: Dynamic binding; Multiple inheritance; Interfaces, abstract classes and generics.
Design, construction and testing

Program specifications
Error handling and exceptions.
Methods of testing, coding practices and basic software engineering techniques.

Learning outcomes

By the end of the module, students should be able to:

- Students should be able to understand the programming process, from the definition of the problem and the design of a solution at an abstract level, to the coding itself with an integrated approach to testing for correctness.
- Student should be able to develop programs in a high-level programming language using the imperative paradigm.
- Students should be able to structure complex software using principles of encapsulation and abstraction in the object-oriented programming abstractions.
- Students should be able to apply informal reasoning techniques to justify the correctness of methods and programs, and justify desired properties such as termination.
- Students should be able to use different strategies for testing programs and managing run-time errors using exception handling.

Indicative reading list

Please see Talis Aspire link for most up to date list.

View reading list on Talis Aspire

Subject specific skills

Experience with the Java programming language
Understanding of Data Types
Iterative Statements
Conditional Statements
Imperative Programming
Object Oriented Programming
Inheritance
Abstract classes

Generics
Multi-threading in Java
Error Handling & Exceptions
Passing by value vs passing by reference.
Learning to think programatically and algorithmically and how to take a specification and turn it into a plan for a program.
Introduction to good coding practices & basic software engineering techniques

Transferable skills

Coding
Problem solving
Communication skills (verbal)
Critical thinking

Study

Study time

Type	Required
Lectures	20 sessions of 1 hour (13%)
Practical classes	10 sessions of 1 hour (7%)
Private study	120 hours (80%)
Total	150 hours

Private study description

A significant proportion of independent study should be spent with the coursework assignments, and additional programming practice. The rest should be revision of the material and practising the concepts discussed.

Costs

No further costs have been identified for this module.

Assessment

You do not need to pass all assessment components to pass the module.

Students can register for this module without taking any assessment.

Assessment group D5

	Weighting	Study time
Programming assignment 1	20%	
This assessment is eligible for self-certification (extension)		
Programming assignment 2	20%	
This assessment is eligible for self-certification (extension)		
In-person Examination	60%	
Exam		

- Answerbook Pink (12 page)

Assessment group R3

	Weighting	Study time
In-person Examination - Resit	100%	
Resit Exam		

- Answerbook Pink (12 page)

Feedback on assessment

Written feedback via Tabula.

Past exam papers for CS118

Availability

Courses

This module is Core for:

- UCSA-G500 Undergraduate Computer Science
  - Year 1 of G500 Computer Science
  - Year 1 of G500 Computer Science
- UCSA-G503 Undergraduate Computer Science MEng
  - Year 1 of G500 Computer Science
  - Year 1 of G503 Computer Science MEng
  - Year 1 of G503 Computer Science MEng
- Year 1 of UCSA-I1N1 Undergraduate Computer Science with Business Studies

- Year 1 of UCSA-G406 Undergraduate Computer Systems Engineering
- Year 1 of UCSA-G408 Undergraduate Computer Systems Engineering
- USTA-G302 Undergraduate Data Science
    - Year 1 of G302 Data Science
    - Year 1 of G302 Data Science
- Year 1 of USTA-G304 Undergraduate Data Science (MSci)
- UCSA-G4G1 Undergraduate Discrete Mathematics
    - Year 1 of G4G1 Discrete Mathematics
    - Year 1 of G4G1 Discrete Mathematics
- Year 1 of UCSA-G4G3 Undergraduate Discrete Mathematics