WM245-18 Programming Languages for Cyber Security

23/24

Department
WMG
Level
Undergraduate Level 2
Module leader
Henry Caushi
Credit value
18
Module duration
30 weeks
Assessment
Multiple
Study location
University of Warwick main campus, Coventry

Description

Introductory description

Programming languages are the medium through which programmers precisely describe concepts, formulate algorithms, and reason about solutions. In the course of a career, a cyber professional may work with many different languages, separately or together. If developing software, they must understand the programming models underlying different languages and make informed design choices in languages supporting multiple complementary approaches. The cyber professional is likely to need to learn new languages and programming constructs. They therefore must understand the principles underlying how programming language features are defined, composed, and implemented. The effective use of programming languages, and appreciation of their limitations, also requires a basic knowledge of programming language translation and static program analysis, as well as run-time components such as memory management.

This module enables students to develop insight into the significant differences between different programming paradigms. Object oriented programming, functional programming and event driven programming are given special emphasis. Students develop solutions under these different paradigms to embed theoretical concepts into professional practice.

Module aims

The module aims to equip students with an understanding of a number of different programming paradigms. We will consider how the choice of language or paradigm can introduce specific opportunities or vulnerabilities into software. We will also explore the factors that can influence programming language or paradigm choice, and how to keep our knowledge of software vulnerabilities up to date.

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.

Outline content
The content of this module will be taught from a cyber security perspective.

- object-oriented programming
- functional programming
- event-driven and reactive programming
- type systems
- program representation
- language translation and execution
- syntax analysis
- compiler semantic analysis
- code generation

Learning outcomes

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

- Compare different programming paradigms used to create software.
- Reflect on how software vulnerabilities can be minimised during software creation.
- Incorporate security features in small-scale programs.
- Develop small-scale programs that employ the idioms of a programming paradigm in a conventional manner.

Indicative reading list

Aho, A. V., Lam, Monica S., Sethi, R. and Ullman, Jeffrey D., “Compilers: Principles, Techniques, and Tools”, 2 Ed, Pearson (2013)

Bird, Richard, “Thinking Functionally with Haskell”, Cambridge University Press (2014)

Friedman, Daniel P. and Wand, Mitchell, “Essentials of Programming Languages”, 3 Revised Ed, MIT Press (2008)

Sarcar, V., "Java Design Patterns", 2nd Edition, Apress (2019)

Subject specific skills

Compare different programming paradigms used to create software.
Reflect on how software vulnerabilities can be minimised during software creation.
Incorporate security features in small-scale programs.
Develop small-scale programs that employ the idioms of a programming paradigm in a conventional manner.

Transferable skills

Problem solving

Study

Study time

Type	Required
Supervised practical classes	18 sessions of 2 hours 30 minutes (50%)
Private study	45 hours (50%)
Total	90 hours

Private study description

Independent activity between workshops, following up on activities initiated in previous workshops or preparing for upcoming workshops.

Costs

No further costs have been identified for this module.

Assessment

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

Assessment group A2

	Weighting	Study time
Assignment	40%	35 hours
A reflection covering programming skills and cyber security.		
Assignment	60%	55 hours

	Weighting	Study time
A report describing a cyber security tool created during the module, and justifying the programming design decisions made during the creation process.		

Assessment group R1

	Weighting	Study time
Coursework (Resit)	100%	

Feedback on assessment

Written feedback for each assignment
Verbal feedback during tutorial sessions
Summative feedback on assignments and practical tests

Availability

Courses

This module is Core for:

- UWMA-H651 Undergraduate Cyber Security
    - Year 2 of H651 Cyber Security
    - Year 2 of H651 Cyber Security
    - Year 2 of H651 Cyber Security