



Course Outline (Higher Education)

School:	School of Engineering, Information Technology and Physical Sciences
Course Title:	FOUNDATIONS OF PROGRAMMING
Course ID:	ITECH1400
Credit Points:	15.00
Prerequisite(s):	Nil
Co-requisite(s):	Nil
Exclusion(s):	(ITECH1000 and ITECH5000)
ASCED:	020103

Description of the Course:

This course provides students with a basic understanding of the fundamental programming concepts that are inherent in all computer programs. In addition an introduction is given to the principles involved in designing, developing, testing and debugging applications for information systems.

Grade Scheme: Graded (HD, D, C, P, MF, F, XF)

Work Experience:

No work experience: Student is not undertaking work experience in industry.

Placement Component: No

Supplementary Assessment: Yes

Where supplementary assessment is available a student must have failed overall in the course but gained a final mark of 45 per cent or above and submitted all major assessment tasks.

Program Level:

Level of course in Program	AQF Level of Program					
	5	6	7	8	9	10
Introductory	<input type="checkbox"/>	<input type="checkbox"/>	✓	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Intermediate	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Advanced	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Learning Outcomes:

Knowledge:

- K1.** Identify and use the correct syntax of a common programming language.
- K2.** Recall and use typical programming constructs to design and implement simple software solutions.
- K3.** Reproduce and adapt commonly used basic algorithms.
- K4.** Explain the importance of programming style concepts (documentation, mnemonic names, indentation).

Skills:

- S1.** Utilise pseudocode and/or algorithms as a major program design technique.
- S2.** Write and implement a solution algorithm using basic programming constructs.
- S3.** Demonstrate debugging and testing skills whilst writing code.
- S4.** Describe program functionality based on analysis of given program code.

Application of knowledge and skills:

- A1.** Develop self-reliance and judgement in adapting algorithms to diverse contexts.
- A2.** Design and write program solutions to identified problems using accepted design constructs.

Course Content:

Topics may include:

- Overview of software development and where programming fits in.
- Problem-solving techniques, program types and programming languages.
- The use of variables, operators and programming syntax.
- Program logic including the use of branching, loops and identifying logic errors.
- Procedures and functions.
- Introduction to the use of data, data persistence and file input/output.
- Main features of procedural programming.
- Introduction to common software development methodologies.

- Searching and sorting techniques.

Values:

- V1.** Develop a professional attitude to the design and implementation of software solutions.
- V2.** Develop problem-solving skills and self-reliance in a program development context.

Graduate Attributes

The Federation University FedUni graduate attributes (GA) are entrenched in the [Higher Education Graduate Attributes Policy](#) (LT1228). FedUni graduates develop these graduate attributes through their engagement in explicit learning and teaching and assessment tasks that are embedded in all FedUni programs. Graduate attribute attainment typically follows an incremental development process mapped through program progression. **One or more graduate attributes must be evident in the specified learning outcomes and assessment for each FedUni course, and all attributes must be directly assessed in each program**

Graduate attribute and descriptor		Development and acquisition of GAs in the course	
		Learning Outcomes (KSA)	Assessment task (AT#)
GA 1 Thinkers	Our graduates are curious, reflective and critical. Able to analyse the world in a way that generates valued insights, they are change makers seeking and creating new solutions.	K1, K2 & K3, A1, A2	1, 2
GA 2 Innovators	Our graduates have ideas and are able to realise their dreams. They think and act creatively to achieve and inspire positive change.	K2, K4, S4	1, 2
GA 3 Citizens	Our graduates engage in socially and culturally appropriate ways to advance individual, community and global well-being. They are socially and environmentally aware, acting ethically, equitably and compassionately.	A1, A2	1, 2
GA 4 Communicators	Our graduates create, exchange, impart and convey information, ideas, and concepts effectively. They are respectful, inclusive and empathetic towards their audience, and express thoughts, feelings and information in ways that help others to understand.	K4	1, 2
GA 5 Leaders	Our graduates display and promote positive behaviours, and aspire to make a difference. They act with integrity, are receptive to alternatives and foster sustainable and resilient practices.	N/A	N/A

Learning Task and Assessment:

Participation in lectures, tutorials and computer laboratory classes. Completion of all tutorial and laboratory worksheets for the semester. Students are expected to spend time regularly out of scheduled classes by reading reference material as required, reviewing topics already covered in lectures and preparing for forthcoming topics and laboratory classes, and completing assessment tasks. Assessment for this course will be based on a number of tasks. These may include written assignments, programming tasks and laboratory exercises covering the systems development and programming design. An end of semester test is based on all aspects of the course.

Learning Outcomes Assessed	Learning Tasks	Assessment Type	Weighting
K1, K2, K3, K4, S1, S2, S3, S4, A1, A2	Students will utilise pseudocode, basic programming constructs, algorithms, design constructs and standard style concepts to design, write, implement and document solutions to simple problems.	Assignments and exercises	65 - 75 %
K1, K2, K3, K4, S2, S3, S4	Students will provide theoretical answers and provide practical solutions to a range of questions and problem types drawn from theory and examples used during this course.	Test(s)	25 - 35%

Adopted Reference Style:

APA

Refer to the [library website](#) for more information

Fed Cite - [referencing tool](#)