

INTRODUCTORY PROGRAMMING

900084

2020



UNIT OUTLINE

Last amended:	August 2020
---------------	-------------

© Western Sydney University Enterprises Pty Limited 2020

Except as provided by the Copyright Act 1968, no part of this publication may be produced, stored in a retrieval system or transmitted in any form or by any means without the prior written permission of Western Sydney University Enterprises Pty Limited.

This unit outline is to be used for educational purposes only.

Students should not make this publication available commercially, or redistribute it by any technological means.

Students must not record lectures or tutorials via any technology unless they obtain the express consent of the lecturer prior to the lecture being given.

Western Sydney University The College Nirimba Education Precinct Eastern Road Quakers Hill NSW 2763

Postal address: PO BOX 224 Quakers Hill NSW 2763

Phone: (02) 9852 4488 Fax: (02) 9852 4480

Disclaimer

If you buy or use this publication you should understand clearly that it has been produced solely for learning purposes. While the author and Western Sydney University Enterprises Pty Limited have made every effort to ensure that the material in this publication is accurate and of high quality, you are expressly advised that you should not rely on the contents of this publication in order to make decisions having legal, accounting, property, financial, investment or similar consequences or for any purpose other than learning. For any purposes other than learning you should first obtain the advice of an appropriately qualified professional. The author and Western Sydney University Enterprises Pty Limited disclaim any liability to any person, whether a student or otherwise, in respect of anything, and the consequences of anything, done or omitted to be done by any such person in reliance, whether wholly or partially, on the whole, or any part of, or omission from, the contents of this publication.

Western Sydney University ABN 53 014 069 881 is a registered provider under the Commonwealth Register of Institutions and Courses for Overseas Students (CRICOS). Provider Number 00917K.

Western Sydney University Enterprises Pty Limited ABN 44 003 474 468 trading as Western Sydney University The College (CRICOS Provider Number 02851G) is a wholly owned entity of Western Sydney University. Academic Pathway Programs are delivered by Western Sydney University The College under arrangement with Western Sydney University.

Unit name	Introductory Programming				
Unit number	900084				
Coordinator	Zdenka Misanovic				
Session	2020.3				
Handbook summary	This unit introduces students to the principles required for the effective design and development of computer programs. This unit has been developed to help students acquire an understanding of essentials in designing programs theoretically and implementing them practically, using an integrated development environment (IDE).				
Credit point value	10				
Prerequisite/s	N/A				
Corequisite/s	N/A				
Unit incompatible with and not to be counted for credit with	N/A				
Assumed knowledge	The ability to create a mathematical expression for a given problem scenario. This would require knowledge of basic arithmetic, percentages and simple statistical measures.				
Unit level	Level Z — Non-award unit				
Attendance requirements	Students are expected to attend at least 80% of classes. Educational research consistently demonstrates that this attendance level is associated with a high likelihood of achieving a passing grade.				
Enrolment restrictions	Students must be enrolled at The College.				
Learning outcomes	 On successful completion of this unit, students should be able to: define terms such as structured programming, variables, constants, control structures, modularisation, cohesion, coupling, function, procedures, parameters and arguments illustrate the steps involved in program development solve problems and illustrate solutions using sequence control structure with flowcharts and/or pseudocode, which are then coded in a 3GL language such as C++ solve problems and illustrate solutions using selection control structure flowcharts, pseudocode and translate to C++ solve problems and illustrate solutions using iteration control structure flowcharts, pseudocode and translate to C++ solve problems using modularisation with parameter passing code, debug and test programs in C++ using an Integrated Development (IDE) and 				
	 develop a set of input test data and desk check pseudocode. 				

Unit content	In this unit students will learn about:				
	 introduction to steps in program design and development 				
	 problem solving and developing algorithms using IPO charts, flow charts and pseudocode 				
	sequence control structure				
	 selection control structures using IF and CASE statements 				
	 repetition control structures using REPEAT, WHILE and FOR loops 				
	 modularisation, procedures/void functions, functions and parameter passing 				
	 input output data tables and desk check tables 				
	 use of an Integrated Development Environment (IDE), and 				
	translating pseudocode into C++ using correct syntax.				
Mode of delivery	This unit consists of six hours of supervised computer laboratory sessions. In addition, students will be required to access vUWS regularly, in order to download additional learning material, and to check for any announcements about the unit that may be posted there.				
Online learning requirements					
Online learning requirements Essential	Essential texts				
Online learning requirements Essential requirements	Essential texts Allain, A 2012, <i>Jumping into C++</i> , CProgramming.com, San Francisco, CA.				
Online learning requirements Essential requirements	Essential texts Allain, A 2012, <i>Jumping into C++</i> , CProgramming.com, San Francisco, CA. The College, <i>Introductory programming workbook</i> , Western Sydney University The College, Sydney.				
Online learning requirements Essential requirements	Essential texts Allain, A 2012, <i>Jumping into C++</i> , CProgramming.com, San Francisco, CA. The College, <i>Introductory programming workbook</i> , Western Sydney University The College, Sydney. Further resources				
Online learning requirements Essential requirements	 Essential texts Allain, A 2012, Jumping into C++, CProgramming.com, San Francisco, CA. The College, Introductory programming workbook, Western Sydney University The College, Sydney. Further resources Farrell, J 2013, Programming logic and design: comprehensive version, 7th edn, Course Technology, Boston, Mass. 				
Online learning requirements Essential requirements	 Essential texts Allain, A 2012, Jumping into C++, CProgramming.com, San Francisco, CA. The College, Introductory programming workbook, Western Sydney University The College, Sydney. Further resources Farrell, J 2013, Programming logic and design: comprehensive version, 7th edn, Course Technology, Boston, Mass. Robertson, LA 2006, Simple program design: a step-by-step approach, 5th edn, Cengage Learning, Australia, South Melbourne, Vic. 				
Online learning requirements Essential requirements	 Essential texts Allain, A 2012, Jumping into C++, CProgramming.com, San Francisco, CA. The College, Introductory programming workbook, Western Sydney University The College, Sydney. Further resources Farrell, J 2013, Programming logic and design: comprehensive version, 7th edn, Course Technology, Boston, Mass. Robertson, LA 2006, Simple program design: a step-by-step approach, 5th edn, Cengage Learning, Australia, South Melbourne, Vic. Essential equipment 				

ASSESSMENT ITEMS AND WEIGHTING

Assessment for this unit will be based on the following components:

Task	Weighting	Learning outcomes assessed	Mandatory task
1. Class work	20%	3-8	Yes
2. Class test (1.5 hours)	20%	3–5	Yes
 Programming project (approx. 100 lines of code) 	20%	3-8	Yes
4. End of Session exam (2 hours)	40%	1–8	Yes
TOTAL	100%		

For details of assessment due dates, please refer to the learning guide for this unit.

All marks will be determined in accordance with The College Assessment Policy.

All assessment tasks are mandatory unless otherwise specified. Should a student fail to attempt/submit the first formal assessment task in a unit, they will be deemed to be at risk and will need to follow an intervention plan in order not to receive a Fail Non-Submission (FNS) grade. However, failure to attempt/submit all other mandatory assessment tasks will result in an immediate FNS grade for the unit.

In order to pass this unit, students must:

- attempt/submit all mandatory assessment tasks including the End of Session exam, and
- achieve a minimum overall mark of 50%.