



CURRICULUM GUIDE: OFFICIAL COURSE OUTLINE

Course Code	CPSC 111	Course Title	Introduction to Computing			
Credit Value	4	Department	Mathematics and Science			
No. of weeks	14	Hrs. per week	<i>Lecture</i>	<i>Tutorial</i>	<i>Laboratory</i>	<i>Total</i>
			3	0	2	5
Course Description	Basic programming constructs, data types, classes, and the design of programs as interacting software components.					
Prerequisite(s)	ENGL 088, MATH 12 (C) or MATH 099 (C) or MATH 100					
Initial Articulation Targets	<i>UBC</i>	<i>SFU</i>	<i>UVic</i>	<i>UNBC</i>	<i>TRU</i>	
	CPSC 1 st (4)	CMPTS 120 (3)	CSC 110 (1.5)	CPSC 110 (3)	COMP 1110 (3)	
	For updated information on the transferability of this course, please consult the BC Transfer Guide, www.bctransferguide.ca					
Learning Outcomes	<p>Upon successful completion of this course, the student will be able to:</p> <ul style="list-style-type: none"> • Write pseudo-code and diagrams to describe the steps of an algorithm • Design, implement, and debug a program that uses input/output, conditional and iterative structures, functions, and objects including class inheritance • Write programs that include common list and string-processing algorithms • Write programs with decision structures, loops, and functions • Design, document, implement, and test solutions to programming problems • Debug programs • Explain basic searching and sorting algorithms • Define the qualities of a good algorithm 					
Content	<p>Core topics – all of the following will be covered:</p> <p>Programming in Python:</p> <ul style="list-style-type: none"> • Algorithms and Computational Thinking • Procedural Programming • Data Types and Control Structures • List and String Processing • Classes, Encapsulation, and Information Hiding • Basic Algorithms including Searching and Sorting, Introduction to Recursive Algorithms • Basic Computability and Complexity <p>Additional topics may also be covered, at the discretion of the instructor:</p> <ul style="list-style-type: none"> • Inheritance 					



Methods of Instruction	Lectures, assignments, computer laboratory work, projects, assigned reading, quizzes, examinations		
Required Textbook(s)	The following textbook(s) is/are required, or approved equivalent(s). Allen, Downey. Think Python-How to Think Like a Computer Scientist-Python 3. Green Tea Press, 2016.		
Required Equipment and Technology	Students are required to have a computer with internet access. The following resources are provided by the College: <ul style="list-style-type: none"> • Office 365 • Student email 		
Homework Hours	At minimum, students can expect one hour of homework for every hour of instructional time.		
Evaluation	<i>Component</i>	<i>% Value</i>	
	Assignments and quizzes	15-30%	
	Labs and projects	15-20%	
	Midterm examinations (1-2)	20-35%	
	Final examination	30-35%	
Completion Requirements	The minimum grade to pass this course is D (50%). Unless otherwise stated, a minimum grade of C- (55%) is required for this course to fulfil a prerequisite.		
Course Designer(s)	Gordon Simon, M.Sc., Computer Science faculty, Alexander College	Consultant(s), if applicable	Tiko Kameda, Ph.D., Professor Emeritus, Department of Computing Science, Simon Fraser University
Dean's Approval	Marv Westrom, Ph.D. Professor Emeritus, Faculty of Education, University of British Columbia	Dean's Approval Date	January 23, 2006
Curriculum Committee Approval Date	January 23, 2006	First Term Offered	Fall 2007
Last Review Date	September 28, 2022	Next Review Date	September 28, 2027
Revision History	<p>April 16, 2007-Revised by Alexandra Fedorova, SFU</p> <p>June 15, 2007-Revised by Alexander Fedorova, SFU</p> <p>December 1, 2008-Revised by Gordon Simon and Tiko Kameda</p> <p>September 28, 2022—Revision of some learning outcomes, content, textbook, assessment by Kelly Cheung.</p>		