



Systematic Program Design – COMP 110

University Studies Program

Course Outline

COURSE IMPLEMENTATION DATE: September 2020

OUTLINE EFFECTIVE DATE: September 2020

COURSE OUTLINE REVIEW DATE: April 2025

GENERAL COURSE DESCRIPTION:

This course introduces students to principles of program design and to implementing and understanding computer programs using an object-oriented programming language. Programming topics include selection and iteration, arrays and collections, objects and classes, top-down design and incremental development. The programming skills developed in this course will help students appreciate program design as a tool for information processing, simulation and modelling, and interacting with the world.

Program Information: This course is an important foundation of many science programs including Physics, Chemistry, Mathematics, Computing Science, and Astronomy.

Delivery: This course is delivered face-to-face.

COTR Credits: 3

Hours for this course: 90 hours

Typical Structure of Instructional Hours:

Instructional Activity	Duration
Lecture Hours	45
Seminars / Tutorials	
Laboratory / Studio Hours	45
Practicum / Field Experience Hours	
Other Contact Hours	
Total	90

Practicum Hours (if applicable):

Type of Practicum	Duration
On-the-job Experience	N/A
Formal Work Experience	N/A
Other	N/A
Total	

Course Outline Author or Contact:

Trevor Beugeling, MASc

Signature

APPROVAL SIGNATURES:

Department Head
Erin Aasland Hall
E-mail: aaslandhall@cotr.bc.ca

Dean of Business and University Studies
Darrell Bethune
E-mail: bethune@cotr.bc.ca

Department Head Signature

Dean Signature

EDCO

Valid from: September 2020 – April 2025

Education Council Approval Date

COURSE PREREQUISITES AND TRANSFER CREDIT:

Prerequisites: Minimum grade of C+ (65%) in Foundation of Math 10, Workplace Math 10, or equivalent.

Corequisites: N/A

Flexible Assessment (FA):

Credit can be awarded for this course through FA Yes No

Learners may request formal recognition for flexible assessment at the College of the Rockies through one or more of the following processes: External Evaluation, Worksite Assessment, Demonstration, Standardized Test, Self-assessment, Interview, Products/Portfolio, Challenge Exam. Contact an Education Advisor for more information.

Transfer Credit: For transfer information within British Columbia, Alberta and other institutions, please visit <http://www.cotr.bc.ca/Transfer>.

Students should also contact an academic advisor at the institution where they want transfer credit.

Textbooks and Required Resources:

Textbook selection varies by instructor and may change from year to year. At the Course Outline Effective Date the following textbooks were in use:

How to Design Programs, Felleisen, Finder, Flatt, Krishnamurthi, The MIT Press.

Please see the instructor's syllabus or check COTR's online text calculator <http://go.cotr.bc.ca/tuition/tCalc.asp> for a complete list of the currently required textbooks.

LEARNING OUTCOMES:

Upon the successful completion of this course, students will be able to

- apply a systematic design process by writing programs for reasonably complex problems and following the “one task – one function” rule;
 - read function designs and identify different elements;
 - evaluate function elements and overall function design;
 - write code that is easy to read and interpret by others, as well as being properly organized, documented, and tested;
 - demonstrate the relation between information and data by designing data representations for reasonably large and complex problems, and describing the information encoded in that data;
 - identify correspondences between data definitions and programs that operate on that data;
 - identify how changes to a data definition will affect a program;
 - apply abstraction to replace repetitive code from programs they have written, then verify that the new program still solves the original problem;
 - write programs that use existing code and libraries to solve new problems;
 - identify correspondences between non-code models of a program and the program itself by using non-code models in program design; and
 - design, implement, test, and debug programs involving basic computations, conditional and iterative structures, and definition of functions.
-

COURSE TOPICS:

- Primitive data, expressions, constants, and variables
- Conditional and iterative control structures
- Function definitions and passing of parameters
- Strategies for problem-solving and algorithm design
- Interactive graphical programs.
- Testing and debugging strategies
- Atomic Data, Compound Data, Structures, Self-Reference and Reference Data, Lists, Arrays, Loops, Natural Numbers, Strings, Images, Boolean Expressions, Helper Functions. Binary Search Trees, Mutual Reference, Local Expressions and Local Functions, Abstract Functions, Recursion, Searching, Accumulators, Graphs, Mutation

See instructor's syllabus for the detailed outline of weekly readings, activities and assignments.

EVALUATION AND ASSESSMENT:

Assignments	% Of Total Grade
Assignments	15%
Labs	10%
Quizzes and in-class activities	15%
Midterm Exams	35%
Final Exam	<u>25%</u>
Total	100%

Please see the instructor syllabus for specific classroom policies related to this course, such as details of evaluation, penalties for late assignments and use of electronic aids.

Note: You must receive a passing grade in each of the Lab component and the Final Exam component in order to pass the course.

EXAM POLICY:

Students must attend all required scheduled exams that make up a final grade at the appointed time and place.

Individual instructors may accommodate for illness or personal crisis. Additional accommodation will not be made unless a written request is sent to and approved by the appropriate Department Head prior to the scheduled exam.

Any student who misses a scheduled exam without approval will be given a grade of "0" for the exam.

COURSE GRADE:

Course grades are assigned as follows:

Grade	A+	A	A-	B+	B	B-	C+	C	C-	D	F
Mark (Percent)	≥ 90	89-85	84-80	79-76	75-72	71-68	67-64	63-60	59-55	54-50	< 50

A grade of "D" grants credit, but may not be sufficient as a prerequisite for sequential courses.

ACADEMIC POLICIES:

See www.cotr.bc.ca/policies for general college policies related to course activities, including grade appeals, cheating and plagiarism.

COURSE CHANGES:

Information contained in course outlines is correct at the time of publication. Content of the courses is revised on an ongoing basis to ensure relevance to changing educational, employment and marketing needs. The instructor will endeavour to provide notice of changes to students as soon as possible. The instructor reserves the right to add or delete material from courses.