

Effective: Spring 2023

## COURSE INFORMATION

**Course Title:** Introduction to Computer Science & Programming II

**Course Number:** CSCI 125

**Credits:** 4

**Total Weeks:** 14 (Fall, Spring)  
12 (Summer)

**Total Hours:** 65

**Course Level:** ☒ First Year ☐ Second Year  
☐ New ☒ Revised Course  
☐ Replacement Course

**Department:** Computer Science **Department Head:** M. O'Connor

**Former Course Code(s) and Number(s) (if applicable):**  
N/A

**Pre-requisites (If there are no prerequisites, type NONE):** PREC 12 and CSCI 120 or equivalent

**Co-requisite Statement (List if applicable or type NONE):** NONE

**Precluded Courses:** N/A

## COURSE DESCRIPTION

This course is a thorough introduction to computer science and computer programming, suitable for students with some computer science background. It is designed for students who will major in computing science or a related program. Students will learn fundamental concepts and terminology of computer programming and acquire basic programming skills in the Java programming language. Topics covered are: primitive and abstract data types, elementary data structures, fundamental algorithms, algorithm analysis, basic object-oriented programming and software design, specification and program correctness, and historical aspects of computing science.

## LEARNING OUTCOMES

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

- Cite some historical facts of computer science and programming.
- Demonstrate how to choose primitive data types appropriately.
- Know the difference between primitive data types and reference data types.
- Describe and apply techniques to document programs.
- Use appropriate tools for software development
- Develop, test, and evaluate programs
- Use good and defensive programming style
- Demonstrate how to use control structures appropriately.
- Demonstrate how to use text files for input and output.
- Know how to construct safe programs.
- Explain and use simple abstract data types such as list, stack and queue.
- Demonstrate how to design and use classes.
- Demonstrate how to use modeling tools such as UML, for object-oriented design.
- Develop object-oriented programming solutions
- Know data structures from the Java Collections API such as arrays, lists and dictionaries.
- Demonstrate how to use single and multi-dimensional arrays.
- Know and analyze fundamental searching and sorting algorithms.
- Explain and apply the concepts of Object-Oriented Programming.
- Define and use recursive algorithms for problem solving.

## INSTRUCTION AND GRADING

Instructional (Contact) Hours:

| Type                      | Duration  |
|---------------------------|-----------|
| Lecture                   | 39        |
| Seminars/Tutorials        |           |
| Laboratory                | 26        |
| Field Experience          |           |
| Other ( <i>specify</i> ): |           |
| <b>Total</b>              | <b>65</b> |

**Grading System:** Letter Grades ☒ Percentage ☐ Pass/Fail ☐ Satisfactory/Unsatisfactory ☐ Other ☐

**Specify passing grade:** 50%

**Evaluation Activities and Weighting** (total must equal 100%)

|                                                                             |                                                                      |                  |                                                                          |
|-----------------------------------------------------------------------------|----------------------------------------------------------------------|------------------|--------------------------------------------------------------------------|
| Assignments: 5%<br><i>Review Questions: 3%</i><br><i>Review Quizzes: 2%</i> | Lab Work: 15%                                                        | Participation: % | Projects: 15%<br><i>Midterm Project: 5%</i><br><i>Final Project: 10%</i> |
| Quizzes/Test: %                                                             | Midterm Exams: 30%<br><i>Midterm 1: 15%</i><br><i>Midterm 2: 15%</i> | Final Exam: 35%  | Other: %                                                                 |

## TEXT(S) AND RESOURCE MATERIALS

Provide a full reference for each text and/or resource material and include whether required/not required.

Tony Gaddis, Godfrey Muganda: Starting Out with Java: From Control Structures through Data Structures, 4th Edition. Pearson. 2019. ISBN 013478796X • 9780134787961.

## COURSE TOPICS

List topics and sequence covered.

| Week   | Topic                                                                     |
|--------|---------------------------------------------------------------------------|
| Week 1 | Introduction to Java, OOP, and Program Design                             |
| Week 2 | Primitive data types and their operations                                 |
| Week 3 | Decision and loop control structures,                                     |
| Week 4 | File input and output                                                     |
| Week 5 | Methods                                                                   |
| Week 6 | <b>Midterm Exam 1</b> / Introduction to Classes and Objects, UML diagrams |

|         |                                                             |
|---------|-------------------------------------------------------------|
| Week 7  | Introduction to Arrays and the ArrayList Class              |
| Week 8  | Sorting and Searching Algorithms, Algorithm Analysis        |
| Week 9  | More about Classes and Objects                              |
| Week 10 | <b>Midterm Exam 2</b> / Test Processing and Wrapper Classes |
| Week 11 | Inheritance and Polymorphism                                |
| Week 12 | Exceptions                                                  |
| Week 13 | Recursion / Abstract Data Types                             |
| Week 14 | <b>Final Exam</b>                                           |

**NOTES**

1. Students are required to follow all College policies. Policies are available on the website at: [Coquitlam College Policies](#)
2. To find out how this course transfers, visit the BC Transfer Guide at: [bctransferguide.ca](http://bctransferguide.ca)

**Last Reviewed:** September 2024

**Last Revised:** January 2023