

## SAMPLE COURSE OUTLINE

### Course Code, Number, and Title:

CPSC 1181: Object-oriented Computing

### Course Format:

[Course format may vary by instructor. The typical course format would be:]

Lecture 4.0 h + Seminar 0.0 h + Lab. 2.0 h

**Credits:** 3.0

**Transfer Credit:** For information, visit [bctransferguide.ca](http://bctransferguide.ca)

### Course Description, Prerequisites, Corequisites:

Object-oriented programming (OOP) is a paradigm to design and develop software based on the concept of objects. Students are introduced to the fundamental concepts of programming from an object-oriented (OO) perspective: abstraction; objects; classes and class hierarchies; methods; encapsulation and information hiding; inheritance; polymorphism. Students learn and practice the application of OO design with modeling tools (e.g., class diagrams), container/collection classes, event-driven programming, exception handling, GUI, multi-threading, and networking. The focus is placed on good software engineering principles using a language that supports the OO paradigm.

Prerequisite(s): A minimum "C" grade in CPSC 1150 or 1155; or permission of department. Prerequisites are valid for only three years.

### Learning Outcomes:

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

- Design and implement applications using the Object-Oriented Programming (OOP) methodology
- Describe and employ the concepts of OOP, such as encapsulation, inheritance, polymorphism, abstract classes, interfaces, exceptions and exception handling
- Design, draw, and explain Unified Modelling Language (UML) class diagrams
- Build a program that manipulates data using a collection, such as an ArrayList or vector
- Implement programs using generic programming
- Use a given design pattern to develop and implement an Object-Oriented (OO) solution to a given problem
- Use, design, and write unit tests
- Apply the example of object-oriented design to build programs containing a selection of event-driven programming, GUI, graphics, multi-threading/race condition/synchronization, network programming, lambdas and streams
- Write internal and external documentation that conforms to language conventions and good programming practice

**Instructor(s): TBA**  
**Office: TBA**  
**Office Hours: TBA**

**Phone: (604) 323-XXXX**  
**Email: TBA**

**Textbook and Course Materials:**

[Textbook selection may vary by instructor. An example of texts and course materials for this course might be:]

For textbook information, visit [https://mycampusstore.langara.bc.ca/buy\\_course/listing.asp?selTerm=3|8](https://mycampusstore.langara.bc.ca/buy_course/listing.asp?selTerm=3|8)

*Note: This course may use an electronic (online) instructional resource that is located outside of Canada for mandatory graded class work. You may be required to enter personal information, such as your name and email address, to log in to this resource. This means that your personal information could be stored on servers located outside of Canada and may be accessed by U.S. authorities, subject to federal laws. Where possible, you may log in with an email pseudonym as long as you provide the pseudonym to me so I can identify you when reviewing your class work.*

**Assessments and Weighting:**

**Final Exam** 25%

**Other Assessments** 75%

[An example of other assessments might be:]

Labs 40%

Weekly Practice Activities 10%

Two Midterms 25%

**Grading System:**

Specific grading schemes will be detailed in each course section outline.

*Information unavailable, please consult Department for details.*

**Topics Covered:**

[Topics covered may vary by instructor. An example of topics covered might be:]

- Data Types, Strings, Testing
- UML, Using Objects, Implementing Classes
- Arrays, Array Lists, Enumerated Types
- Inheritance, Polymorphism, Abstract Classes
- Designing Classes, OO Design, UML
- Interfaces, Lambda, Callbacks, Events
- Graphical Applications, Shape Classes
- Comparable Interface, I/O, Exception handling
- GUI: Layout, Swing, MVC
- Multi-threading, Internet Networking
- I/O Streams
- Race Conditions, Deadlocks

**“This Sample Course Outline is for planning purposes only”.**

Page 2 of 3

As a student at Langara, you are responsible for familiarizing yourself and complying with the following policies:

**College Policies:**

[E1003 - Student Code of Conduct](#)

[F1004 - Code of Academic Conduct](#)

[E2008 - Academic Standing - Academic Probation and Academic Suspension](#)

[E2006 - Appeal of Final Grade](#)

[F1002 - Concerns about Instruction](#)

[E2011 - Withdrawal from Courses](#)

**Departmental/Course Policies:**

*Information unavailable, please consult Department for details.*

SAMPLE