



OOP244 - Introduction to Object Oriented Programming

School of Information and Communications Technology, Summer - 2023

Subject Title

Introduction to Object Oriented Programming

Subject Description

This subject introduces the student to object-oriented programming. The student learns to build reusable objects, encapsulate data and logic within a class, inherit one class from another and implement polymorphism. This subject uses the C++ programming language exclusively and establishes a foundation for learning system analysis and design and more advanced concepts as implemented in languages such as C++, Java, C# and Objective-C.

Credit Status

1 credit (3 units)

Required for CPA - Computer Programming and Analysis (Ontario College Advanced Diploma)

Required for CPD - Computer Programmer (Ontario College Diploma)

Learning Outcomes

Upon successful completion of this subject the student will be able to:

1. Design classes with dynamically allocated resources to model the components of a programming solution
2. Design member functions using logic constructs to solve tasks of linear complexity
3. Relate classes using inheritance hierarchies to minimize the duplication of object code
4. Design polymorphic objects to amplify the reusability of program code
5. Use stream objects to interact with users and access persistent data
6. Trace the execution of object-oriented code to validate its correctness
7. Code a complete program using polymorphic objects to solve a systems or business problem
8. Explain the purpose of an object-oriented programming feature to inform a business person

Essential Employability Skills

- Execute mathematical operations accurately.
- Apply a systematic approach to solve problems.
- Use a variety of thinking skills to anticipate and solve problems.
- Locate, select, organize, and document information using appropriate technology and information systems.
- Analyze, evaluate, and apply relevant information from a variety of sources.
- Take responsibility for one's own actions, decisions, and consequences.

Academic Integrity

Seneca upholds a learning community that values academic integrity, honesty, fairness, trust, respect, responsibility and courage. These values enhance Seneca's commitment to deliver high-quality education and teaching excellence, while supporting a positive learning environment. Ensure that you are aware of Seneca's Academic Integrity Policy which can be found at: <http://www.senecacollege.ca/about/policies/academic-integrity-policy.html> Review section 2 of the policy for details regarding approaches to supporting integrity. Section 2.3 and Appendix B of the policy describe various sanctions that can be applied, if there is suspected academic misconduct (e.g., contract cheating, cheating, falsification, impersonation or plagiarism).

Please visit the Academic Integrity website <http://open2.senecac.on.ca/sites/academic-integrity/for-students> to understand and learn more about how to prepare and submit work so that it supports academic integrity, and to avoid academic misconduct.

Discrimination/Harassment

All students and employees have the right to study and work in an environment that is free from discrimination and/or harassment. Language or activities that defeat this objective violate the College Policy on Discrimination/Harassment and shall not be tolerated. Information and assistance are available from the Student Conduct Office at student.conduct@senecacollege.ca.

Accommodation for Students with Disabilities

The College will provide reasonable accommodation to students with disabilities in order to promote academic success. If you require accommodation, contact the Counselling and Accessibility Services Office at ext. 22900 to initiate the process for documenting, assessing and implementing your individual accommodation needs.

Camera Use and Recordings - Synchronous (Live) Classes

Synchronous (live) classes may be delivered in person, in a [Flexible Learning space](#), or online through a Seneca web conferencing platform such as MS Teams or Zoom. Flexible Learning spaces are equipped with cameras, microphones, monitors and speakers that capture and stream instructor and student interactions, providing an in-person experience for students choosing to study online.

Students joining a live class online may be required to have a working camera in order to participate, or for certain activities (e.g. group work, assessments), and high-speed broadband access (e.g. Cable, DSL) is highly recommended. In the event students encounter circumstances that impact their ability to join the platform with their camera on, they should reach out to the professor to discuss. Live classes may be recorded and made available to students to support access to course content and promote student learning and success.

By attending live classes, students are consenting to the collection and use of their personal information for the purposes of administering the class and associated coursework. To learn more about Seneca's privacy practices, visit [Privacy Notice](#).

Prerequisite(s)

IPC144 and APS145

Topic Outline

- Introduction - 5%
 - Complexity, Languages, Namespaces
 - Object Terminology
 - Abstraction, Objects and Classes
 - Encapsulation, Inheritance and Polymorphism
 - Modular Programming

- Modules
 - Stages of Compilation
 - Unit Tests
 - Debugging Techniques
- Foundations - 20%
 - Types, Overloading and References
 - Declarations and Definitions
 - Scope
 - Function Overloading
 - References
 - Arrays of Pointers
 - Dynamic Memory
 - Allocation and Deallocation - new and delete
 - Memory Issues
 - Member Functions and Privacy
 - Member Functions
 - Privacy
 - Simple Input and Output Examples
 - Standard Input and Output
 - Member Functions
- Encapsulation - 30%
 - Constructors and Destructors
 - Classes and Privacy
 - Construction and Destruction of Arrays
 - Overloading Constructors
 - The Current Object
 - Member Function Parameters
 - this
 - Member Operators
 - Operations - unary and binary
 - Type Conversion Operations
 - Casting Operations
 - Temporary Objects
 - Classes and Resources
 - Resource Instance Pointers
 - Copy Constructor
 - Assignment Operator
 - Member Function Delet
 - Helper Functions
 - Free Helpers (non-friends)
 - Helper Operators
 - Friendship
 - Input and Output Operators
 - Stream Library Overview
 - Custom Standard Operators
 - Custom File Operators
 - string class
- Inheritance - 10%
 - Derived Classes
 - Hierarchies
 - Base and Derived classes
 - Access - public, protected, private
 - Functions in a Hierarchy
 - Shadowing
 - Constructors and Destructors
 - Helper Operators
- Polymorphism - 20%
 - Virtual Functions
 - Types
 - Early Binding and Dynamic Dispatch

- Polymorphic Objects
- Abstract Base Classes
 - Pure Virtual Functions
 - Arrays of Pointers
 - Unit Tests on an Interface
- Templates
 - Function Templates
 - Class Templates
 - Constrained Casts
- Overview of Polymorphism
 - Languages and Type Systems
 - Categories - coercion, overloading, inclusion, parametric
- Refinements - 10%
 - Simple Input and Output Examples
 - Input Objects
 - Output Objects
 - Manipulators
 - State
 - Robust Validation
 - File Stream Classes
 - Derived Classes and Resources
 - Copy Constructor
 - Copy Assignment Operator
 - Language Standards
 - History - pre-standard C++, C++98, C++11, C++14
 - Some Features that have changed
 - Differences between C++ and C

Mode of Instruction

Modes: In-class lecture, in-class exercises, and hands-on activity

Hours per week: 4

Room configurations: Classroom, and computer lab

Typical scheduling pattern: Fall, Winter, and summer terms

Prescribed Texts

Hard Copy | Introduction to Object-Oriented Programming Using C++14, July 2016 Edition

by Chris M. Szalwinski

published by Seneca College

Available in the bookstore

Online | Introduction to Object-Oriented Programming Using C++14

<https://scs.senecac.on.ca/~oop244>

by Chris M. Szalwinski

Reference Material

OOP244 Web Notes by Chris Szalwinski

<https://scs.senecac.on.ca/~oop244>

Required Supplies

None

Student Progression and Promotion Policy

- Must satisfactorily complete project
- Must have weighted passing average of mid-term and final
- Must have overall passing average

<http://www.senecacollege.ca/about/policies/student-progression-and-promotion-policy.html>

Grading Policy<http://www.senecacollege.ca/about/policies/grading-policy.html>

A+	90% to 100%
A	80% to 89%
B+	75% to 79%
B	70% to 74%
C+	65% to 69%
C	60% to 64%
D+	55% to 59%
D	50% to 54%
F	0% to 49% (Not a Pass)
OR	
EXC	Excellent
SAT	Satisfactory
UNSAT	Unsatisfactory

For further information, see a copy of the Academic Policy, available online (<http://www.senecacollege.ca/about/policies/academics-and-student-services.html>) or at Seneca's Registrar's Offices. (<https://www.senecacollege.ca/registrar.html>).

Modes of Evaluation

Workshops – 25%

Project – 15%

Quizzes – 15%

Mid-term – 15%

Final Test – 30%

Approved by: Kathy Dumanski

Last revision date: 30-Jan-2023 12:32:34 AM

Last review date: 3-Apr-2023 12:15:04 AM