

CS321

Advanced Programming Techniques

Course Information

- Session: Winter 2023
- Instructor: Christopher Diggins
- Email: cdiggins@ubishops.ca
- Office: Johnson 103-A
- Lectures: Tuesdays/Thursdays 15:00-16:30 Hamilton 151
- Labs: Tuesdays/Thursdays 16:30-18:00 Johnson 118
- Office Hours: by appointment

Class Description

This class is a sequel to an CS211 Introduction to Programming. You will learn advanced programming techniques such as object-oriented programming, functional programming, and software architectural patterns in the context of the C# language.

The first half of the course will focus on core features and concepts of C#, object-oriented programming, test-driven development, and functional programming. The second half of the course will focus on more advanced programming techniques, as well as principles, practices, and patterns of software design and architecture.

The language of instruction will be C# 7.0 and the programming environment will be Visual Studio on Windows. The techniques, principles, and patterns learned will be applicable to various high-level multi-paradigm languages and programming environments.

The lab assignments will emphasize building familiarity with C#, Visual Studio, .NET, and coding techniques, with the aim of preparing students to be able to effectively complete the final project. Examples include using the debugger, release and debug builds, unit and integration testing, refactoring code, code reuse, using packages, code analysis, using Git and GitHub, and disassemblers.

Topics Covered

The following are the major topics to be covered during this class

- Core Language Features and Evaluation Model of C#
- The .NET Systems Library
- Object Oriented Programming and Design (Polymorphism, Inheritance, Encapsulation, etc.)
- Interfaces and Extension Methods
- Functional Programming
- Collections and Generics
- Unit Tests and Test-Driven Development

- Reflection and Serialization
- Exceptions and Error Handling
- Event Systems
- Data Binding
- Asynchronous and Concurrent Programming
- Software Architecture Principles and Patterns (e.g., SOLID, MVC, MVVM, etc.)

Texts

There is no required reading. Class notes and assignments will be posted online at <https://github.com/cdiggins/cs321>.

Class notes will reference official online source: C# Language Reference Online: <https://learn.microsoft.com/enus/dotnet/csharp/language-reference>

Suggested supplemental reading: C# in Depth, 4th Edition by Jon Skeet. <https://www.manning.com/books/c-sharp-in-depth-fourth-edition>

Course Evaluation

- Assignments (5) – 20%
- Labs (8) – 25%
- Midterm (Closed book) – 15%
- Final Project (Non-group) – 40%

Late Policy

No in class, or lab assignments, will be accepted after the deadline, however I will drop the mark of the lowest assignment and the lowest lab.

Midterm

The midterm will only cover material that has been covered in class, and that can be found both in notes and that online in the official language reference documentation. The midterm will be closed book.

Final Project

The final project details will be revealed after the midterm. It will be a solo project of a non-trivial piece of software that you will post on GitHub.

Plagiarism

Plagiarism is a kind of academic dishonesty in which an individual uses the work of another without appropriate acknowledgement. Plagiarism includes but is not limited to the following practices:

- Using another's work without acknowledgement
- Copying material without quotation marks
- Paraphrasing too closely the exact words of the originating author
- Submitting work written in whole or in part as one's own by another individual

Your submitted work must be your own creation.

The following is considered acceptable if you credit all sources, including friends, and AI:

- Using an AI tool for help (like GitHub Copilot)
- Using StackOverflow.com
- Using ReSharper
- Asking friends for help

The key is to do the work yourself, use other sources as tools.