Introduction

CS 115

Dr. Joseph Eremondi, adapted from Dr. Shakil Khan, Dr. Philip Fong, and Dr. Howard Hamilton

Last updated: January 6, 2025

Welcome to CS 115

What is this course all about?

Objectives

What is this course all about?

Objectives

- Various concepts of object oriented programming. Topics include: two-dimensional arrays, records, data abstraction, classes, composition and inheritance, type systems, subtyping, dynamic binding, polymorphism, pointers/references, dynamic memory management, and searching and sorting algorithms. Along the way, we may also discuss software engineering concepts, including comprehensibility, correctness, efficiency, and refactoring.

What is this course all about?

Major theme: Types

What is this course all about?

Major theme: Types

- This course gives you your first real look at *types* in programming

What is this course all about?

Major theme: Types

- This course gives you your first real look at *types* in programming
  - How to define your own types

What is this course all about?

Major theme: Types

- This course gives you your first real look at *types* in programming
  - How to define your own types
  - How to use types to structure programs

What is this course all about?

Major theme: Types

- This course gives you your first real look at *types* in programming
  - How to define your own types
  - How to use types to structure programs
- More than just `int` and `bool`

What is this course all about?

Major theme: Software Engineering

What is this course all about?

Major theme: Software Engineering

- CS 110 is how to write code

What is this course all about?

Major theme: Software Engineering

- CS 110 is how to write code
- CS 115 is how to write **good** code

What is this course all about?

Major theme: Software Engineering

- CS 110 is how to write code
- CS 115 is how to write **good** code
- i.e. How to write code that is easy to:

What is this course all about?

Major theme: Software Engineering

- CS 110 is how to write code
- CS 115 is how to write **good** code
- i.e. How to write code that is easy to:
    - Understand

What is this course all about?

Major theme: Software Engineering

- CS 110 is how to write code
- CS 115 is how to write **good** code
- i.e. How to write code that is easy to:
  - Understand
  - Maintain

What is this course all about?

Major theme: Software Engineering

- CS 110 is how to write code
- CS 115 is how to write **good** code
- i.e. How to write code that is easy to:
  - Understand
  - Maintain
  - Re-use

What is this course all about?

Major theme: Software Engineering

- CS 110 is how to write code
- CS 115 is how to write **good** code
- i.e. How to write code that is easy to:
  - Understand
  - Maintain
  - Re-use
  - Extend

What is this course all about?

Major theme: Software Engineering

- CS 110 is how to write code
- CS 115 is how to write **good** code
- i.e. How to write code that is easy to:
  - Understand
  - Maintain
  - Re-use
  - Extend
- *Intentionally limiting* the ways we are allowed to use parts of our program

What is this course all about?

Major theme: Software Engineering

- CS 110 is how to write code
- CS 115 is how to write **good** code
- i.e. How to write code that is easy to:
  - Understand
  - Maintain
  - Re-use
  - Extend
- *Intentionally limiting* the ways we are allowed to use parts of our program
  - Abstraction is at the heart of computer science

Tentative Outline

- Review

Tentative Outline

- Review
- 2D arrays, records (structs), abstract data types

Tentative Outline

- Review
- 2D arrays, records (structs), abstract data types
- Searching and sorting

Tentative Outline

- Review
- 2D arrays, records (structs), abstract data types
- Searching and sorting
- Object Oriented Design

Tentative Outline

- Review
- 2D arrays, records (structs), abstract data types
- Searching and sorting
- Object Oriented Design
    - Classes and constructors

Tentative Outline

- Review
- 2D arrays, records (structs), abstract data types
- Searching and sorting
- Object Oriented Design
  - Classes and constructors
  - Overloading and coercion

Tentative Outline

- Review
- 2D arrays, records (structs), abstract data types
- Searching and sorting
- Object Oriented Design
  - Classes and constructors
  - Overloading and coercion
  - Composition and inheritance

Tentative Outline

- Review
- 2D arrays, records (structs), abstract data types
- Searching and sorting
- Object Oriented Design
  - Classes and constructors
  - Overloading and coercion
  - Composition and inheritance
- Pointers and dynamic memory management

Tentative Outline

- Review
- 2D arrays, records (structs), abstract data types
- Searching and sorting
- Object Oriented Design
  - Classes and constructors
  - Overloading and coercion
  - Composition and inheritance
- Pointers and dynamic memory management
- Program organization

Learning Objectives

By the end of this course, you should be understand

Learning Objectives

By the end of this course, you should be understand

- Data abstraction, encapsulation, polymorphism

Learning Objectives

By the end of this course, you should be understand

- Data abstraction, encapsulation, polymorphism
- Solving real world problems with object-oriented principles

Learning Objectives

By the end of this course, you should be understand

- Data abstraction, encapsulation, polymorphism
- Solving real world problems with object-oriented principles
- Using composite types and abstract types to solve real-world problems

Learning Objectives

By the end of this course, you should be understand

- Data abstraction, encapsulation, polymorphism
- Solving real world problems with object-oriented principles
- Using composite types and abstract types to solve real-world problems
- Approaches to searching and sorting

Learning Objectives

By the end of this course, you should be understand

- Data abstraction, encapsulation, polymorphism
- Solving real world problems with object-oriented principles
- Using composite types and abstract types to solve real-world problems
- Approaches to searching and sorting
- C++ OOP features

Learning Objectives

By the end of this course, you should be understand

- Data abstraction, encapsulation, polymorphism
- Solving real world problems with object-oriented principles
- Using composite types and abstract types to solve real-world problems
- Approaches to searching and sorting
- C++ OOP features
    - Operator overloading, type coercions, constructors, destructors

Learning Objectives

By the end of this course, you should be understand

- Data abstraction, encapsulation, polymorphism
- Solving real world problems with object-oriented principles
- Using composite types and abstract types to solve real-world problems
- Approaches to searching and sorting
- C++ OOP features
  - Operator overloading, type coercions, constructors, destructors
- Pointers, dynamic memory management, and linked data structures

Learning Objectives

By the end of this course, you should be understand

- Data abstraction, encapsulation, polymorphism
- Solving real world problems with object-oriented principles
- Using composite types and abstract types to solve real-world problems
- Approaches to searching and sorting
- C++ OOP features
  - Operator overloading, type coercions, constructors, destructors
- Pointers, dynamic memory management, and linked data structures
- Modular programming with inheritance and composition

Grading

- 20% assignments

Grading

- 20% assignments
- 15% labs

Grading

- 20% assignments
- 15% labs
- 20% midterm

Grading

- 20% assignments
- 15% labs
- 20% midterm
- 45% final

Grading

- 20% assignments
- 15% labs
- 20% midterm
- 45% final
  - Must pass final to pass the class

Midterm

- In-class, Friday March 7

Midterm

- In-class, Friday March 7
  - Might be split into two rooms

Labs

- See https://www.labs.cs.uregina.ca/115/

Assignments

- 4 Assignments

Assignments

- 4 Assignments
- Submitted on URCourses

Assignments

- 4 Assignments
- Submitted on URCourses
- Sample based marking

Assignments

- 4 Assignments
- Submitted on URCourses
- Sample based marking
    - You submit the whole thing, we mark part of it

Office Hours

- Tuesdays 11:00-12:00

Office Hours

- Tuesdays 11:00-12:00
- Thursdays 10:30-11:30

Office Hours

- Tuesdays 11:00-12:00
- Thursdays 10:30-11:30
- In RIC 317

Office Hours

- Tuesdays 11:00-12:00
- Thursdays 10:30-11:30
- In RIC 317
    - Take the elevator then go across the bridge

Office Hours

- Tuesdays 11:00-12:00
- Thursdays 10:30-11:30
- In RIC 317
  - Take the elevator then go across the bridge
  - or, take the stairs by the vending machines

Course Communication

- In lectures

Course Communication

- In lectures
- Announcements on URCourses

Course Communication

- In lectures
- Announcements on URCourses
- Course email on URCourses

Course Communication

- In lectures
- Announcements on URCourses
- Course email on URCourses
  - For privately contacting instructor

Course Communication

- In lectures
- Announcements on URCourses
- Course email on URCourses
  - For privately contacting instructor
  - Save for things that need to be private

Course Communication

- In lectures
- Announcements on URCourses
- Course email on URCourses
  - For privately contacting instructor
  - Save for things that need to be private
    - Personal circumstances

Course Communication

- In lectures
- Announcements on URCourses
- Course email on URCourses
  - For privately contacting instructor
  - Save for things that need to be private
    - Personal circumstances
    - Assignment solutions

Course Discussion Forum

- Ask questions on URCourses!

Course Discussion Forum

- Ask questions on URCourses!
  - Can be asked anonymously

Course Discussion Forum

- Ask questions on URCourses!
  - Can be asked anonymously
  - If you’re wondering, then other students probably are too

Course Discussion Forum

- Ask questions on URCourses!
  - Can be asked anonymously
  - If you’re wondering, then other students probably are too
  - Don’t post partial or complete assignment solutions on the forum

Course Discussion Forum

- Ask questions on URCourses!
  - Can be asked anonymously
  - If you’re wondering, then other students probably are too
  - Don’t post partial or complete assignment solutions on the forum
- Good for

Course Discussion Forum

- Ask questions on URCourses!
  - Can be asked anonymously
  - If you’re wondering, then other students probably are too
  - Don’t post partial or complete assignment solutions on the forum
- Good for
  - Clarification on assignments

Course Discussion Forum

- Ask questions on URCourses!
  - Can be asked anonymously
  - If you're wondering, then other students probably are too
  - Don't post partial or complete assignment solutions on the forum
- Good for
  - Clarification on assignments
  - Understanding course material

Course Discussion Forum

- Ask questions on URCourses!
  - Can be asked anonymously
  - If you're wondering, then other students probably are too
  - Don't post partial or complete assignment solutions on the forum
- Good for
  - Clarification on assignments
  - Understanding course material
  - General curiosity/information beyond the lectures

Academic Honesty

Students are expected to complete assignments *independently*

Academic Honesty

Students are expected to complete assignments *independently*

- No sharing solutions

Academic Honesty

Students are expected to complete assignments *independently*

- No sharing solutions
- No copying from the internet

Academic Honesty

Students are expected to complete assignments *independently*

- No sharing solutions
- No copying from the internet
- No using ChatGPT, Copilot, Claude, or any other Generative AI tool.

Academic Honesty

Students are expected to complete assignments *independently*

- No sharing solutions
- No copying from the internet
- No using ChatGPT, Copilot, Claude, or any other Generative AI tool.

You need to pass the final, so set yourself up for success and do the assignments

Other Logistics

- Attendance expected

Other Logistics

- Attendance expected
  - You’re responsible for anything you miss

Other Logistics

- Attendance expected
  - You’re responsible for anything you miss
- There are detailed course notes on URCourses

Other Logistics

- Attendance expected
  - You're responsible for anything you miss
- There are detailed course notes on URCourses
  - by Howard Hamilton and Phillip Fong

Other Logistics

- Attendance expected
  - You're responsible for anything you miss
- There are detailed course notes on URCourses
  - by Howard Hamilton and Phillip Fong
  - Excellent **free** resource

Other Logistics

- Attendance expected
  - You're responsible for anything you miss
- There are detailed course notes on URCourses
  - by Howard Hamilton and Phillip Fong
  - Excellent **free** resource
- Lectures are a great time to **ask questions**