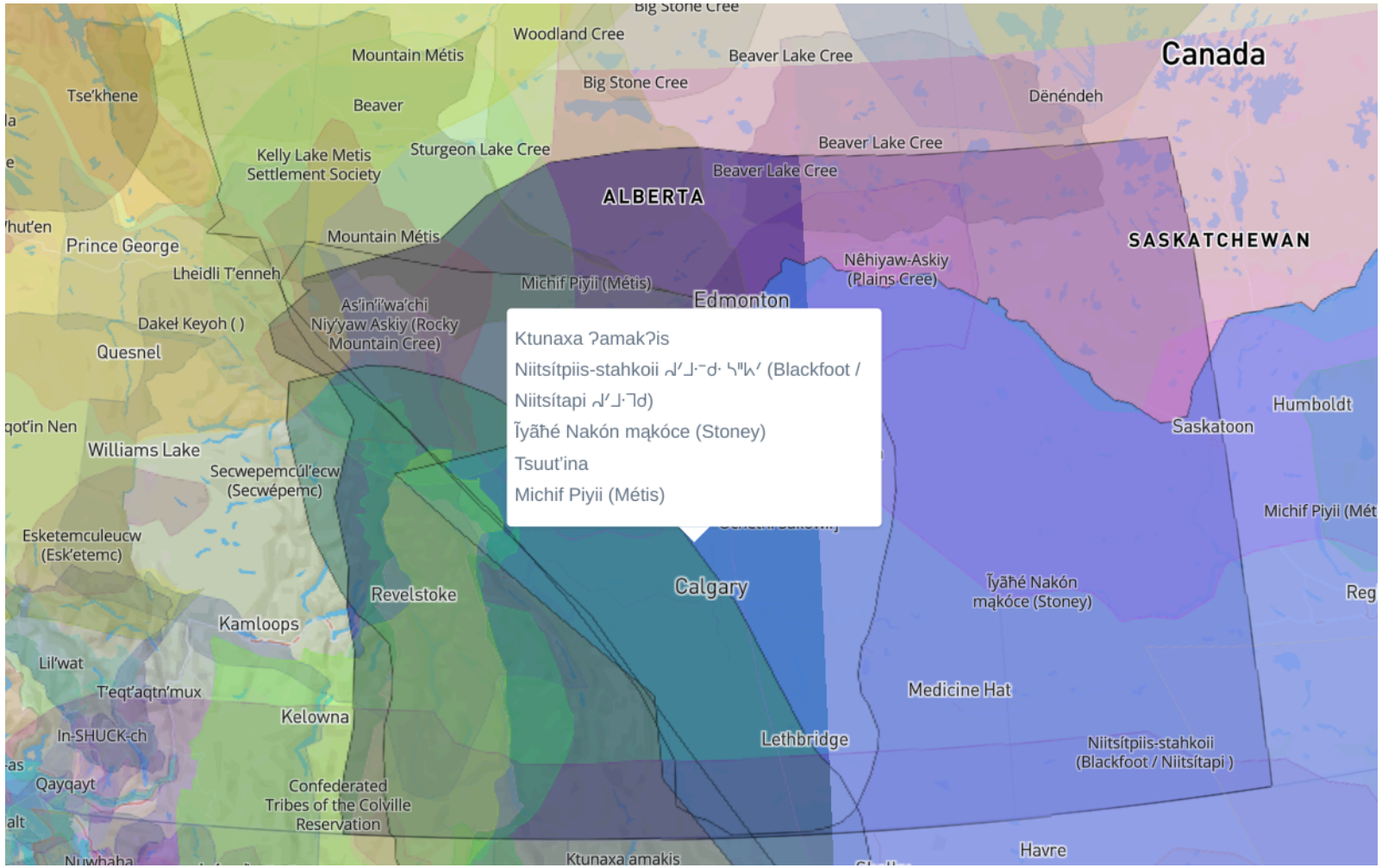
COMP 1633: Intro to CS II

Intro to the course and C++

Charlotte Curtis

January 8, 2024

Land Acknowledgement



Source: https://native-land.ca

Today's topics

- Course intro: assessment structure, policies, tools, etc
- Dive in to C++!

About me

Name: Charlotte Curtis

Background: Biomedical Engineering undergrad, Electrical Engineering PhD

Research: Vector graphics and PDF manipulation (sewing patterns)

Pronouns: She/her

Call me: Charlotte

Office: B175P

Email: ccurtis@mtroyal.ca

What is this class all about?

> *Introduction to object-oriented analysis and design, programming using an object-oriented language, and implementation of linked data structures. Issues of modularity, software design, and programming style will be emphasized.*

- Continuing our problem solving focus from Programming I, but in C++
- Introduction to lower level concepts like memory management
- Introduction to object-oriented programming

Why another programming language?

Python

- Intuitive syntax
- Allows you to focus on the **algorithm**
- Lots of magic, like garbage collection

C++

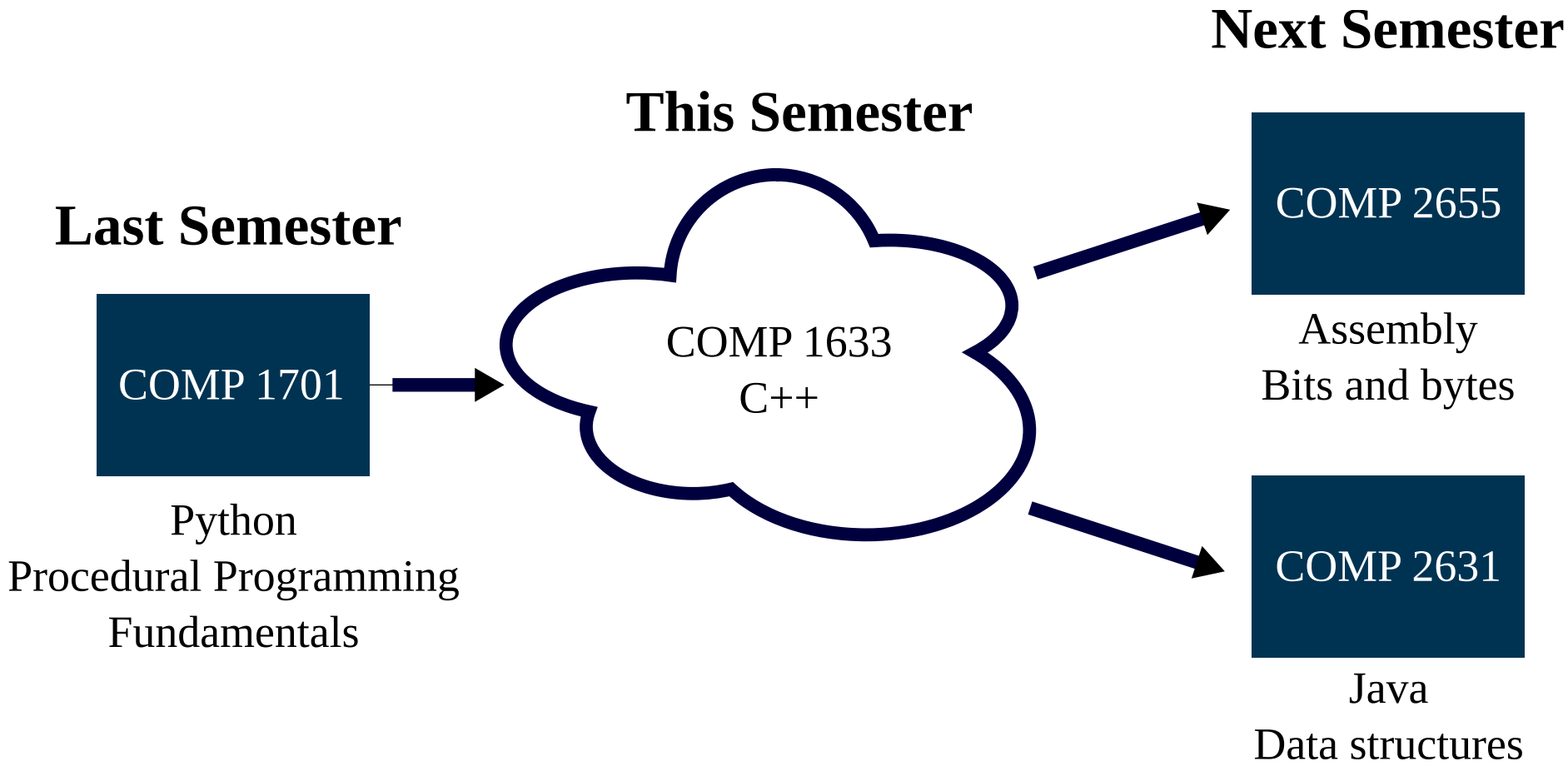
- More complex syntax
- Makes you to think about **memory management**
- Closer to the metal

> *As CS students, you will learn a lot about how computers work and different ways of interacting with them*

Course objective highlights

- Solve problems of **moderate complexity** and magnitude
- Design solutions using classes and other **complex data structures**
- Design and implement programs in C++ on a **headless system**
- Develop and debug **large programs** in a systematic manner
- Explain the concepts and develop programs using:
  - pointers and dynamic memory allocation
  - linked data structures
  - recursion
  - classes and objects
  - lists, stacks, and queues

Where does this course fit?



What is this class *not* about?

> *"C makes it easy to shoot yourself in the foot; C++ makes it harder, but when you do it blows your whole leg off" - Bjarne Stroustrup*

- This is not a C++ course, we're just using C++ as a tool
- This is not a course on modern C++, or even "best practices in C++"
- We are going to violate many rules of good C++ programming in order to:
  - Recognize when you've shot yourself in the foot and what to do about it
  - Develop an appreciation for the protections provided by modern languages

The year was 1998

- A company named "Google" was founded
- Apple released the iMac, saving the company from bankruptcy
- Almost 25% of Canadian Households had internet access
- The first C++ standard was released...
- ...and we're going to use it!

Image credit:
https://www.reddit.com/r/windows98/comments/155szb2/low_end_windows_98_build/

Assessments

Assessment	Weight	Description
Lab exercises	10%	
Assignments	32%	4 assignments, 8% each
Midterm	20%	80 minutes, March 6th, 2024
Final	38%	3 hours, during final exam period

Course Format

- **Lectures** (3 hours per week): Introduction to new concepts, demos, info dump
- **Tutorials** (2 hours per week): Hands-on practice in the lab
- **Assignments**: Projects to be completed outside of class time

> *Attendance is not mandatory, but highly correlated to success*

Lab Exercises

- Exercises during each tutorial, but 15 graded labs for a total of 10%
- Autograded on INS, with manual upload to D2L weekly
- You are encouraged to work together to solve these problems, but must submit individually
- Strongly encouraged to complete during lab time, but you have one week to submit each one

Assignments

- 8% each for a total of 32%
- **Start working on them early!** A portion of each assignment mark is set aside for "evidence of incremental development"
- These assignments are expected to take a significant amount of time, but working on assignments is a great way to "study" for exams
- You have a total of 4 late days to use on assignments throughout the semester

Assignment late policy

- You have a total of 4 days in a "late bank" that can be used to submit assignments late
- Late bank can be used in increments of 0.5 days
- Once your late bank is exhausted, **no late assignments will be accepted**
- **You must indicate in your D2L submission** when you are choosing to use your late bank

Image credit: https://commons.wikimedia.org/wiki/File:Piggy_Bank_On_Pennies_(5915295831).jpg

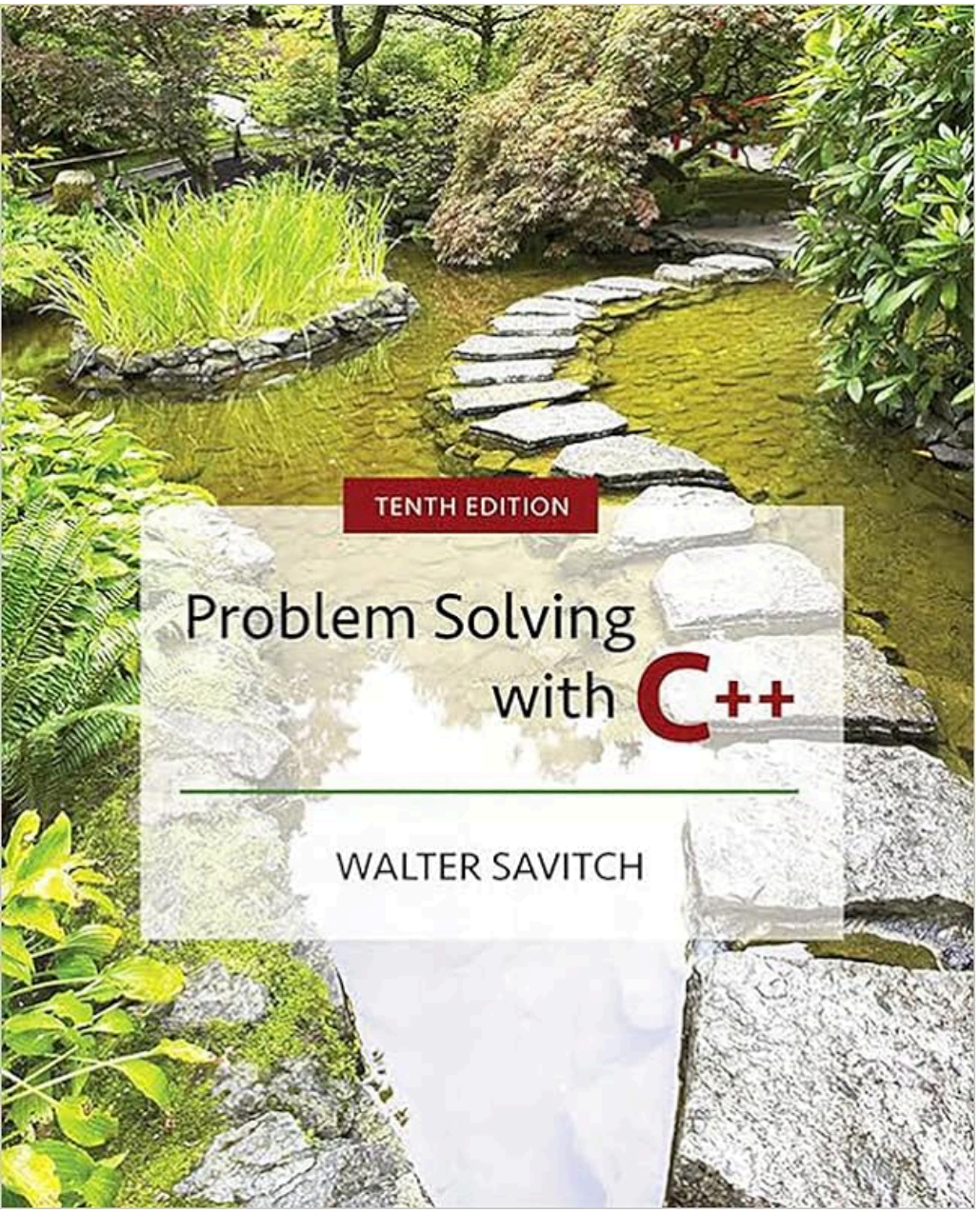
Academic Integrity

- As deadlines start piling up, it can be tempting to copy an assignment
- Both **copying** and **allowing your work to be copied** are considered academic misconduct and will be **reported**
- Your submissions will be compared for similarity using compare50
- If you use an internet resource (e.g. Stack Overflow), **cite it**
  - Just drop the link in your code as a comment
  - ChatGPT/Copilot can be used as informational resources, but straight copying is **not allowed**

> *If you read something, understand it, and can implement it **without looking at the source**, you're not in violation of the academic integrity policy*

Getting Help

- Instructional assistants Jordan and Steve
  - In person in B103A/B107A, or online
- Me (Charlotte):
  - Office B175-P, ccurtis@mtroyal.ca
  - I try to answer emails within 1 business day, generally not on weekends
- Student learning services: mru.ca/sls
  - Mentorship, webinars, personal appointments, etc
  - Accommodations
- CAMRU Discord - join the COMP 1633 study group under channels and roles



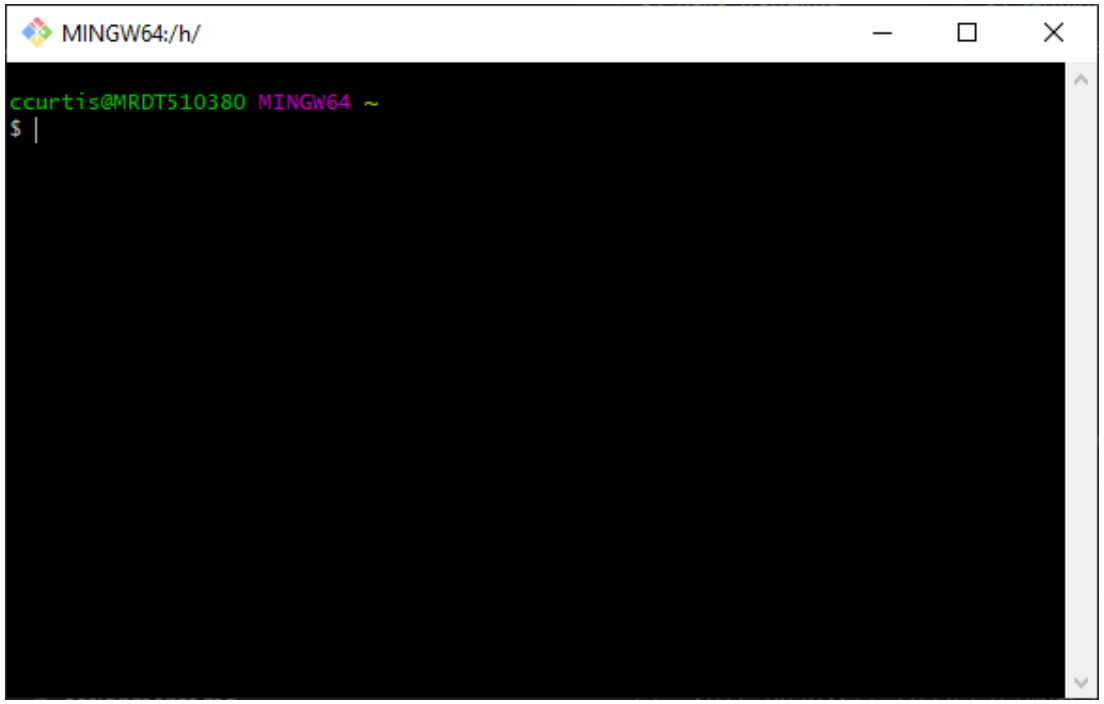
Course resources

- D2L: Grades, announcements, assignment instructions, links
- iClicker: Interactive quizzes/polls
- **Textbook**: Problem solving with C++ by Walter Savitch
  - Optional, but a great resource
  - 9th edition is fine

Development Tools

- Git Version control system
- Emacs Text editor
- C++ compiler: g++ for C++ 98

> *We will be working on a **Linux server** called INS*



Tangent: Let's talk about Linux

- When you see the symbol, I'll be doing an iClicker activity
- Go to join.iclicker.com and enter the code on the board

What comes to mind when you hear the word "Linux"?

Hello World

Python

```python
print("Hello World!")
```

C++

```cpp
#include <iostream>
using namespace std;

int main() {
    cout << "Hello World!\n";
    return 0;
}
```

> *To save space on slides, I will be omitting `#include <iostream>` and `using namespace std;` most of the time. However, these are needed to compile!*

Tracing time!

- Trace the code to the right, predict what is printed to the terminal, and submit your answer in iClicker
- Feel free to discuss with your neighbours

```
int main() {
    int x = 0;
    int z = 0;
    while (x < 5) {
        z += x * x;
        ++x;
    }
    cout << z << '\n';
    return 0;
}
```

Coming up Next

- Lab: Emacs and Git on INS
- Lecture: C++ Basics

> *Textbook Sections 1.3-1.4, 2.1-2.5*