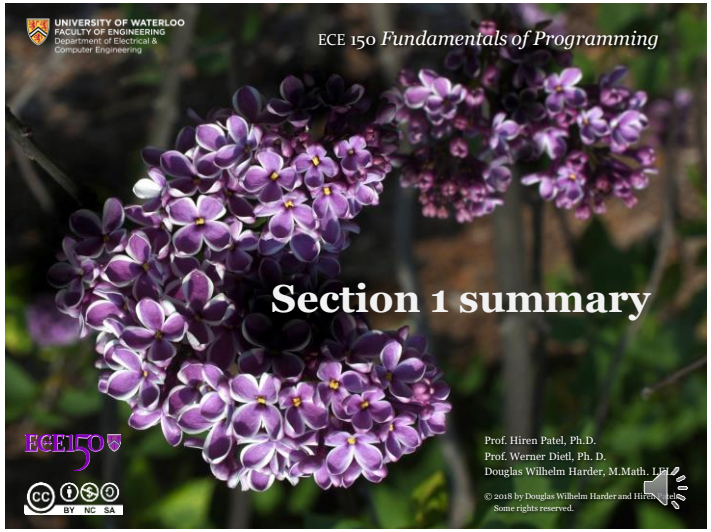

Outline

- In this lesson, we will:
  - Review what has been covered in this topic
  - Look ahead to the next topics

Review

- Up to this point, we have
  - Seen the basics of programming, including
    - Literal data and types
      - Integer, floating-point, character and Boolean types
    - Identifiers
    - Local variables and arrays
    - Arithmetic and logical operators
    - Assignment and auto-assignment operators
    - Conditional and looping statements
      - For loops and while loops
    - Functions, parameters and return values
    - Libraries
    - Binary and hexadecimal representations of numbers
    - Bitwise and bit-shift operators
    - Pass-by-value and pass-by-reference
    - The `const` keyword

Review

- We have also seen a few additional topics:
  - The structured programming theorem
  - Some basic programming and code-development strategies
    - Assertions
    - Code skeletons
    - Tests
    - Comments
    - Logging
    - Tracing
  - The structure of memory, including
    - The code segment
    - The data segment
    - The call stack
  - The use of the call stack for function calls
  - C-style strings

Next topic

- Now that we have discussed memory and addresses
  - Next we will look at storing addresses in variables
    - Such local variables and parameters are called *pointers*
  - The use of addresses in the dynamic allocation of memory
    - Memory that exists outside the scope of a function

Subsequent topics

- Following this, we will see algorithms, including
  - Describing *sorted* arrays
  - Searching algorithms on sorted arrays
  - Sorting algorithms
  - Recursive algorithms
- Next we will investigate classes and their features
  - We will focus on linked lists, contrasting them with arrays

Summary

- Following this topic, you now
  - Have seen a brief review of the topics we have seen to this point:
    - Most deal with the details of programming languages
    - Some aspects covering the development of code
  - Understand at the next topic:
    - Addresses, pointers and dynamic memory allocation
  - Know where we will go from here:
    - Algorithms, classes and specifically linked lists

References

None so far.

Colophon

These slides were prepared using the Georgia typeface. Mathematical equations use Times New Roman, and source code is presented using `Consolas`.

The photographs of lilacs in bloom appearing on the title slide and accenting the top of each other slide were taken at the Royal Botanical Gardens on May 27, 2018 by Douglas Wilhelm Harder. Please see

https://www.rbg.ca/

for more information.

Disclaimer

These slides are provided for the ECE 150 *Fundamentals of Programming* course taught at the University of Waterloo. The material in it reflects the authors' best judgment in light of the information available to them at the time of preparation. Any reliance on these course slides by any party for any other purpose are the responsibility of such parties. The authors accept no responsibility for damages, if any, suffered by any party as a result of decisions made or actions based on these course slides for any other purpose than that for which it was intended.