Student Learning Outcomes

1. Students will become familiar with the internal storage of integral data.
2. Students will learn how to create, compile, link, and run a program in a Unix operating environment.
3. Students will learn how to create multi-file source programs.
4. Students will be introduced to bit manipulation, including left and right shift operators and bitwise operators (not, and, or, exclusive or).
5. Students will be introduced to pointers.
6. Students will learn about character data, including its representation and available functions for testing and manipulating characters.
7. Students will be introduced to the string data type and various functions for manipulating strings.
8. Students will be introduced to structured data.
9. Students will be introduced to object oriented programming using the class concept.

Course Content

Textbook: *Starting out with C++ From Control Structures through Objects*, Sixth Edition, by Tony Gaddis

The following chapters including the particular sections listed are covered. (See textbook “Contents”)

J. **Binary Numbers and Bitwise Operators**
   - Converting Base-B Numbers to Base-10
   - Converting Base-10 Whole Numbers to Base-B
   - Converting Base-10 Fractional Numbers to Base-B
   - Converting from Base-B<sub>1</sub> to Base-B<sub>2</sub>
   - Base-B Complements
   - Unsigned, Sign Magnitude, One’s Complement, and Two’s Complement Storage Modes
   - The Bitwise Operators: left shift, right shift, bitwise not, bitwise and, bitwise or, and bitwise exclusive or

9. **Pointers**
   - Getting the Address of a Variable
   - Pointer Variables
   - The Relationship Between Arrays and Pointers
   - Pointer Arithmetic
   - Initializing Pointers
   - Comparing Pointers
   - Pointers as Function Parameters
   - Dynamic Memory Allocation
   - Returning Pointers from Functions

10. **Characters, Strings, and the string Class**
    - Character Testing
    - Character Case Conversion
    - The C++ string class

11. **Structured Data**
    - Abstract Data Types
    - Combining Data into Structures
    - Accessing Structure Members
    - Initializing a Structure
    - Arrays of Structures
13. **Introduction to Classes**

- Procedural and Object-Oriented Programming
- Introduction to Classes
- Defining an Instance of a Class
- Why Have Private Members?
- Separating Class Specification from Implementation
- Inline Member Functions
- Constructors
- Passing Arguments to Constructors
- Destructors
- Overloading Constructors
- Private Member Functions
- Arrays of Objects

**Additional Content**

Any section or chapter not listed previously.