

Topics to Study for Exam I

Below are the learning outcomes from each of the four modules we have covered leading up to Exam I. You should be able to answer each question or perform the action listed. Feel free to ask questions on Piazza if a topic is unclear.

- M1. Primitive Data Types, Variables, & Arithmetic Operators
 - L01. List C++ primitive data types and explain the appropriate use of each data type
 - L02. List & identify C++ arithmetic operators, translate math equations to C++, and solve arithmetic expressions
 - L03. Construct a simple interactive application
 - L04. Describe how a computer generates a program from code
 - L05. Explain where and how variables are stored in memory
 - L06. Discuss how to convert one data type to another
 - L07. Explain how a computer generates random numbers and write a program that generates random numbers
 - L08. Create a program with formatted output.

- M2. Boolean Expressions & Conditionals
 - L01. Identify C++ control structures and conclude which branch a sample program will execute
 - L02. List C++ logic operators and evaluate Boolean expressions consisting of multiple logic operators
 - L03. Evaluate the resultant output of a code block containing a control structure.
 - L04. Generate pseudocode to solve a decision problem
 - L05. Write a program that implements the pseudocode and solves the problem
 - L06. Identify common errors exhibited in code that contains Boolean expressions. Correct errors in Boolean logic that causes the program to not execute as intended.

- M3. Looping (for / while / do-while)
 - L01. Identify C++ repetition structures and explain the following terms: looping parameter, stopping condition, and looping parameter modification.
 - L02. Explain the appropriate use and differences between a while loop, for loop, and a do-while loop.
 - L03. Convert a program written with a for loop to a program using a while loop and vice versa
 - L04. Explain how scope relates to looping structures
 - L05. Evaluate the resultant output of a given code block containing a loop.
 - L06. Generate pseudocode to solve a repetition problem

- L07. Write a program that implements the pseudocode and solves the problem
- L08. Identify and correct errors in looping structure and program logic.
- L09. Discuss the effect and purpose of using break and continue in a loop.
- L010. Solve an expression containing a prefix or postfix increment operation.

M4. Arrays & Algorithms

- L01. Describe the differences between & advantages of an array & vector.
- L02. Describe the differences between & advantages of a c-style string & a string.
- L03. Sketch how an array is stored in memory denoting the base address and element step size
- L04. Sketch how a multidimensional array is stored in memory denoting the base address, row step size, and element step size
- L05. Explain how sorting an array affects the performance of searching for a value in an array
- L06. Evaluate the resultant output of a given code block containing an array, string, or vector.
- L07. Construct a program that accesses an element in a vector/string, accesses an element in a multidimensional array, modifies an element in a vector/string, returns the length of a vector/string, changes the length of the vector/string, and other vector/string operations
- L08. Generate pseudocode to (1) find the minimum & maximum value in an array (2) sort an array using selection sort
- L09. Write a program that implements the corresponding pseudocode using an array
- L010. Identify and correct errors in array structures, element access & modification, and program logic.
- L011. Create string variables and apply string functions to access or modify the string. (length, at, find, substr)
- L012. Create vector variables and apply vector functions to access or modify the vector (size, at, push_back)