

Topics to Study for Exam II

Below are the learning outcomes from each of the three modules we have covered leading up to Exam II. 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.

M5. Functions & Recursion

- L01. Explain the difference between a function prototype and a function definition. Identify the parts of a function.
- L02. Explain the difference between a parameter and an argument for a function. Discuss what value can be returned from a function and what a void function is.
- L03. Explain the difference between pass-by-value and pass-by-reference. Draw a diagram of how each stores its parameters in memory.
- L04. Explain the meaning of the DRY principle and appropriate uses for functions.
- L05. Discuss how functions contribute towards the principle of single responsibility
- L06. Explain the concept of local & global scope when functions are used within a program
- L07. Evaluate the resultant output of a given code block containing a function call.
- L08. Generate pseudocode to implement a sub-function. Generate pseudocode to solve a larger problem using the sub-functions.
- L09. Write a program that implements the pseudocode and solves the problem.
- L010. Explain the meaning of a stopping condition, the base step, and the recursive step.
- L011. Evaluate the resultant output of a given code block containing a recursive function call.
- L012. Generate pseudocode to implement a recursive function.
- L013. Write a program that implements the pseudocode and solves the recursive problem.
- L014. Identify and correct errors in function structure, argument passing, and program logic.

M6. I/O Streams (iostream / fstream)

- L01. Recite the six steps to properly using a file stream for reading and writing
- L02. Discuss various paradigms for reading in structured data from a file
- L03. Explain the two ways to open a file for writing (create/overwrite & append)
- L04. Evaluate the resultant output of a given code block containing a file stream

- L05. Generate pseudocode to solve a problem that requires a REPL (read-eval-print-loop)
 - L06. Write a program that implements the corresponding pseudocode and solves the REPL problem
 - L07. Identify and correct errors in data stream structure and program logic.
- M7. Classes and Object-Oriented Programming
- L01. Explain the following terms and how they are used (1) dot operator / member access operator (2) scope resolution operator (3) data member (4) member function (5) accessor modifiers (6) accessor & mutator functions
 - L02. Discuss the concept of encapsulation
 - L03. Discuss how classes contribute towards the principle of single responsibility
 - L04. Draw a Class Diagram to describe the structure of a class and its members
 - L05. Discuss the difference between a class and an object
 - L06. Discuss the concept of scope within a class
 - L07. Explain how an object is passed to a function.
 - L08. Create a class that contains objects as data members.
 - L09. Write a program that chains the dot operator to access inner data members.
 - L010. Explain the three uses of const in our programs and classes.
 - L011. Discuss what the this keyword refers to when used within a class.
 - L012. Explain what effect marking a data member or member function as static has in a class.
 - L013. Write a class header file and a class definition file that separates the implementation from the prototype that compiles without error
 - L014. Evaluate the resultant output of a given code block containing a class definition and object.
 - L015. Generate pseudocode to solve a problem that can be more efficiently solved using a class
 - L016. Write a program that implements the corresponding pseudocode and solves the class problem
 - L017. Write a program using a third party library (SFML) that makes use of previously written classes *[Note from Dr. Paone – SFML is not on any quiz or exam]*
 - L018. Identify and correct errors in class structure, object member access, file structure, and program logic.