Chapter 0

You will need to know:

* The categories of computers.
* What are the basic things/commands that computers perform.
* What are the types of computer memory.
* What are the parts of a computer:
  + The tasks they perform.
  + What is done in each (timing of tasks).
  + Where the instruction is being held at execution.
* Definitions: What are input and output devices.
* What is an Operating System.
* Characteristics of data (ex: digital, analog, bits, bytes, KB, MB, and GB).
* What are the characteristics of:
  + ASCII
  + EBCDIC
  + Unicode
* What types of programs are involved in creating, compiling and running your C++ program. Examples: Linker, Loader, Compiler, Assembler, etc.
* Steps of problem solving and what is involved with each.
* Definition of an algorithm and how to create one.
* What is an object and its purpose.
* What is a class and its purpose.

**Chapter 1**

You will need to know:

* What is meant by syntax and semantic as they apply to C++.
* Identify what are legal and illegal identifiers. Look at syntax.
* What are valid values of datatypes, such as int, double, char, etc.
* You will have to determine:
  + Values of arithmetic expresions.
  + Values/statements that use incremental operators (++) and decremental operators (--).
    - Example: num1 = ++num2, or num1 = num2++, etc.
  + Values of output of C++ statements.
* New line character \n and how it is used. (Do not get it confused with the null character \0 which is in chapter 6.)
* Where do the various statements(types) belong? Ex: using namespace std; or include.
* What are bugs in relation to C++ programs.
* Walk-throughs – what are they and when is it beneficial to use them.
* Know how to evaluate static\_cast.
  + Example: static\_cast<int>(72.9) + static\_cast<int>(22.1) This evaluates to 94.

**Chapter 2**

You will need to know:

* Determine what input statements are valid and which are invalid.
* Given input for variables, determine what the values for each would be after statement execution. (FOR ALL TYPES.)
* What are prompt lines and their purpose.
* What are directives and their purpose.
* What are manipulators and their purpose.
* What are input statements and their purpose.
* Ensure that you understand how input works: Its characteristics such as using multiple data types, multiple input lines, how given input is read, how newline characters are used (both declared and understood), how whitespace is read / interpreted with certain data types, and any others.
* The ignore function: what it is and how to use.
* You will have to determine the values of strings.
* You will have to determine values (understand how they work) of:
  + Fixed.
  + Showpoint.
  + Setprecision.
* How to define, open, close, and use:
  + Input data files.
  + Output data files.
* Know about the ios::app option, when dealing with files, it adds output to the end of an already existing file. Hint: app stands for append.
* Chapter 0
* You will need to know:

**Chapter 3**

You will need to know:

* What is a relational operator. Know the different ones.
* Know the difference between the assign operator and the equality operator.
* Determine values of expressions when using && and ||.
* What are nested control statements and how to identify them.
* Given examples of statements you will have to determine what will cause a logical error.
* What is the conditional operator, its characteristics, and how is it used.
* You will have to choose the value of variable(s) in complex expressions.
* How to choose the output of code that uses switch and case.
* What are switch and case.
* What are the characteristics of the break statement.
* How are default labels used in switch expressions? Are the always reached? Is it a best practice to use them?

**Chapter 4**

You will need to know:

* What are the characteristics of looping structures.
* What are the different types of looping structures.
* How are while statements used. Know how to determine output using them.
* What are the different types of while statements, there are 4. Know how each of them works and be able to determine output when they are used.
* IMPORTANT: With while statements
  + Know the difference and outcomes if the cout, cin, and other statements or expressions are within the while vs. outside the while.
  + Know the same above in regards to for loops, and do…while loops.
* Be able to identify the parts(statements) of a loop.
* Know the three control statements of a for loop, what are their names, and what is their purpose / how do they work.
* How to determine the values of variable(s) when used in different types of loops.
* What are the functions of the break statement and how is it used.

**Chapter 5**

You will need to know:

* How the different types of function parameters used, and what they contain.
* Know the different types of functions.
* PLEASE REMEMBER:
  + A program is mainly a collection of functions.
  + In int main() – main is a function.
* What are the different parts of a function. Know how to identify if the are valid or not.
* What are the values of compound returns. (Ex. Return 2 4;) They will only return the last value.
* What is a valid function definition. Know how to identify.
* What is a reference parameter? How is it used? How is it coded?
* How calling a function works, i.e. what is the type of parameter being passed from and to what type of parameter.
* What are the different type of identifiers (local, global, etc.) and when they are accessible (ex: inside or outside of a function (block)).
* In reference to variables; how long do they remain allocated (i.e. global, local, etc.).
* What is function overloading, its characteristics, and how it is used.

**Chapter 6**

You will need to know:

* How to determine the output of variables after calling a one-dimensional array.
* How to define and declare an array.
* What part is contained in [ ] when defining an array. The number of components in an array.
* What part is contained in [ ] when calling an array. The index of the component in the array.