

Concepts and Skills for CMIS 102, Introduction to Problem Solving and Algorithm Design
(Revision 3 – 27 June 2009)

| Topic | Concepts | Skills | How to Study |
|---|---|--|---|
| History, Ethics, and Computer Organization | | | |
| History of Computing | Digital Computer Analog Computer Mainframe Personal Computer Key Contributors to the Development of Computers | Explain these terms Know key contributors and what their contributions were. (You are NOT expected to know dates to any resolution finer than a century.) | Venit pp. 1-8 (3 rd and 4 th editions) Module 1 Section I Note 1 |
| Organization of a Computer | Processor Memory Mass Storage Devices Input Devices Output Devices | Explain these terms. Explain how these items are organized to form a computer | Venit pp. 9-13 (9-15 in 3 rd edition) Module II Section I |
| Programming Languages | Machine Language Assembly Language High-Level Language | Explain these terms Know the names of the most important high-level languages, and the origin and application of these languages | Venit pp. 14-17 (16-20 in 3 rd edition) Note 2 |
| Ethical Issues in Computing | Access Control Privacy Security Viruses and other malware Intellectual Property | Explain these terms and discuss related issues. | Module I Section III |
| Basic Programming Concepts | | | |
| Data Types | Boolean Character Integer Float | Explain the appropriate uses, and limitations, of each of these data types Given a type of data in an application area, select an appropriate data type. | Venit pp. 42-60 (50-60 in 3 rd edition) Module 2 Section 4 |
| Constants | Boolean Character Integer Float String Escape sequence | Be able to write constant values in C++, including string constants that include newline, tab, backslash, and quote. | C++ Language Reference for CMIS102 (You can find this under Course Content in WebTycho) |

| Topic | Concepts | Skills | How to Study |
|-------------|---|---|---|
| Identifiers | Identifier syntax | Recognize and write syntactically correct identifiers in C++ | Module II Section III-A C++ Language Reference for CMIS102 |
| Variables | Storage assignment Word length Declaration Initialization | Write declarations for variables of these types in C++, both with and without initialization. | Venit pp. 31-34 (7-8 in 3 rd edition) Module II Section III-B C++ Language Reference for CMIS102 |
| Operators | Arithmetic Operators Relational Operators Logical Operators | Know and use the C++ symbols for specific operators Explain the difference between integer division and real (float) division | Venit 36-38, 119-126 (43-44, 143-144 in the 3 rd edition) Module 2 Section V |
| Expressions | Operator Precedence Use of Parentheses | Given an expression involving operators of various precedence levels, insert parentheses to show the order in which the operations will be performed. Given an expression, data types for each of the variables, and values for each of the variables, compute the value of the expression Write an expression in C++ to carry out a specified computation | Module 2 Section V C++ Language Reference for CMIS102 Note 3 |
| Statements | Statement separation and termination Assignment Statements Compound Statements Elementary Input/Output Comments | Given a C++ program, identify each of the statements that make up the program, including compound statements Given an assignment statement, data types for each of the variables, and values for each of the variables on the right side, determine what value will be assigned to the variable on the left. Write correct assignment statements and compound statements in C++ Recognize and use both the /*...*/ and // comment forms in C++ | Venit 28 (42-43 in 3 rd edition) Module 2 Section VI-A C++ Language Reference for CMIS 102 |

| Topic | Concepts | Skills | How to Study |
|---|--|---|--|
| Control structures | Selection: if...then and if... then...else "For" loops "While" loops "Do...while" loops "Case" statements | Write correct C++ code to implement each of these structures Given a sample selection statement, data types for the variables, and values for the variables, determine which of the subordinate statements will be executed and which will not. Given a sample of loop code, data types for the variables, and initial values for the variables, determine how many times the loop will be executed, and the final values of the variables. | Venit chapters 3-5 (3 and 4 in the 3 rd edition) Module 3 C++ Language Reference for CMIS 102 |
| Input and output | Console input and output | Write simple C++ statements to obtain numbers or characters from the keyboard, and to write numbers, characters and strings to the display | Code provided in the Modules with each of the Programming Exercises C++ Language Reference for CMIS 102 |
| More Advanced Programming Concepts | | | |
| Program Structure and Development | Modular Design Functions Parameters | Given a task, break it down into modules, and draw a module hierarchy diagram to illustrate the breakdown Given a simple function and actual values for each of the parameters, determine what value the function returns Write a simple function in C++ to meet a requirement Write, compile and successfully run a C++ program with at least one function in addition to main. Explain the difference between call by value and call by reference | Venit Chapter 8 (7 in 3 rd edition) Module 4 Sections I through VI |
| Arrays | Arrays Subscripts Two-dimensional arrays Null-terminated strings | Given a range of subscripts (starting at 0), write a C++ declaration for an array admitting that range of subscripts. Write C++ statements to set or retrieve the values of elements of an array. Explain how arrays are used to store strings in C++. | Venit Sections 6.1, 6.4, and 6.5 (5.1 and 5.4 in the 3 rd edition) Module 5 Section I Note 3 |

| Topic | Concepts | Skills | How to Study |
|-----------------------------|---|--|---|
| Object-oriented programming | Classes Objects Attributes Methods Structures | Explain the concepts of class, object, structure, attribute, and method. | Venit Section 9.1 (8.1 in 3 rd edition) Module V Sections II and III |
| Coding and Testing | Coding Practices Testing Methods | Adhere to good practices when writing code Explain the difference between black-box and white-box testing | Venit Sections 2.1 through 2.3 (both editions) Module IV Sections VII through IX |
| Web Pages | HTML | While referring to a suitable reference work or notes, write HTML code for a short page containing simple features | UMUC Programming Guide Sections 7 and 15 |

Notes:

1. Babbage's Analytical Engine, and the Harvard Mark I, were digital computers (not analog).
2. Additional notes on high-level programming languages:
 - BASIC was developed at Dartmouth College in 1964 as a part of the Dartmouth Time-Sharing System, which provided all undergraduate students with access to a computer. Since its inception, dozens of mutually incompatible dialects of BASIC have been developed by various computer manufacturers and software houses.
 - Java was developed by Sun Microsystems. Small Java programs called applets ("little application") can be embedded in Web pages; but Java is also used for development of large commercial applications.
3. The pseudocode in Venit describe 0-based arrays, while the pseudocode in the Course Modules describe 1-based array design. In C++, arrays are 0-based.