COURSE INF1210:	**COMPUTER SCIENCE 1**
Level:	Introductory
Theme:	Computer Science
Prerequisite:	None
Description:	Students are introduced to the nature, approaches and areas of interest of computer science and its relationship to areas, such as computer engineering and information technology. Students explore concepts associated with hardware, software and processes at an introductory level. There is an emphasis on sequential and structured programming approaches.
Parameters:	Designed to be taught in conjunction with INF1080 Programming 1, INF2150 Programming 2, INF1090 Information Highway 1 and INF1070 Hypermedia Tools as a Grade 10 course in Computer Science.

Curriculum and Assessment Standards

General Outcomes	Assessment Criteria and Conditions	Suggested Emphasis
The student will:	*Assessment of student achievement should be based on:*	
• identify and describe the nature, approaches and areas of interest of computer science	• a test, presentation or project designed to address the following topics: – computer science's central focus on the nature and techniques of problem solving – the role of the algorithm as a foundation of the discipline of computer science – the general areas of interest of computer science – the relationship among computer science, computer engineering and information technology – common misconceptions about computer science	10
• explain and demonstrate the nature, developmental process, use of basic algorithms associated with input processing output (IPO) and structured approaches, and application of these idioms to create complex algorithms	• a presentation or project designed to demonstrate: – the basic nature of algorithms – the ability to design, develop and explain IPO (sequential) and structured algorithms – proficient use of key basic algorithms (idioms) and the ability to use these idioms to create other, complex algorithms	20

COURSE INF1210: COMPUTER SCIENCE 1 (continued)

General Outcomes	Assessment Criteria and Conditions	Suggested Emphasis
The student will:	*Assessment of student achievement should be based on:*	
• explain and demonstrate the nature, evolution, types and role of programming languages	• a presentation or project designed to explain and demonstrate: – the basic nature, evolution, types and role of programming languages	20
• explain and demonstrate the rationale, three fundamental control structures and representation of data in sequential and structured programs	• a presentation or project demonstrating: – the translation of algorithms into structured programs – the basic nature, approach and representation of data in sequential and structured programs	30
• explain the nature, evolution and basic architecture of the von Neumann computer system	• a presentation or project addressing the evolution and nature of the von Neumann computer architecture under the direction of a simple program to explain: – the nature of the five main hardware "blocks" of the computer – a number of typical devices that make up each block – the flow of data through each block of the computer – the relationship with the basic data-processing paradigm. *Assessment Tool* *Assessment Checklist: Computer Science 1 Concepts (Introductory), INF1210–1* *Sample Assignments: Computer Science 1 Concepts (Introductory), INF1210–2* *Standard* *Rating of 2 for the Problem-solving Phase and a rating of 3 for the Implementation Phase*	20
• demonstrate basic competencies.	• observations of individual effort and interpersonal interaction during the learning process. *Assessment Tool* *Basic Competencies Reference Guide and any assessment tools noted above*	Integrated throughout

COURSE INF1210: COMPUTER SCIENCE 1 (continued)

Concept	Specific Outcomes	Notes
Nature of Computer Science	*The student should:* • discuss the following topics: – computer science's central focus is the study of the nature and techniques of problem solving with a particular interest in problems that are solvable by computation – the algorithmic approach is used to solve problems – computer systems are used to test/implement algorithmic solutions to problems – algorithms are used to develop generalized applications useful for solving classes of problems • describe the general areas of interest of computer science. They include: – development and analysis of algorithms – computing systems and their components – communication—both human/machine and machine/machine – formal languages—natural and artificial – automata – artificial intelligence – general development of IT applications • compare and explain computer science versus computer engineering and information technology – theoretical versus applied – general versus specific – exploratory versus applicatory • describe some of the misconceptions associated with computer science: – is the study of computer systems – is synonymous with programming – is the learning of various computer applications	.

COURSE INF1210: COMPUTER SCIENCE 1 (continued)

Concept	Specific Outcomes	Notes
Algorithmic Problem Solving	*The student should:* • describe how an algorithm: – is a step-by-step set of instructions that results in a solution to a problem – becomes a computer program when expressed in a programming language • demonstrate iterative and incremental approaches in the analysis and design stages of the software development process • carry out the first two steps of the Systems Development Life Cycle (Analysis and Design) using: – flowcharts – pseudocode – IPO charting • demonstrate a number of core algorithms, such as: – accumulation (keeping a running total) – determining the mean – determining minimums and maximums	
Implementing the Algorithm (Software and Software Development)	• demonstrate the third step of the Systems Development Life Cycle (Development or Coding) using iterative and incremental approaches • demonstrate the nature of programming language; specifically, that these languages: – reflected a simplified version of natural language • grammar • syntax • semantics • imperative vocabulary • statements • blocks • evolved in tandem with algorithms and hardware • through 5 “generations” • a continuum from machine language to natural language programming	

COURSE INF1210: COMPUTER SCIENCE 1 (continued)

Concept	Specific Outcomes	Notes
	The student should: • each successive generation closer to natural or human language • each generation requires more sophisticated translation into a machine understandable form (assemblers, compilers, interpreters) • "higher" generation languages easier for humans to use but slower and less machine efficient • first generation: machine language • second generation: assembly language • third generation: high-level languages • fourth generation: computer-assisted programming languages • fifth generation: natural language programming – reflected IPO or the data processing paradigm • initialization • input statements • processing statements • output statements • termination/linking • demonstrate how programming languages dealt with data representation: – binary and hexadecimal systems – standard data types – data storage • demonstrate structured programming concepts: – rationale for structured programming – goto-less programming – three fundamental control structures (sequential, decision and iterative) • demonstrate iterative and incremental approaches in the Implementation and Maintenance stages of the Systems Development Life Cycle	

Concept	Specific Outcomes	Notes
Executing the Algorithm (Computer Systems)	*The student should:* • demonstrate computer architecture by producing/explaining/describing: – a block diagram of a stereotypical von Neumann machine • input block or stage • processing block or stage • output block or stage • internal storage block or stage (memory) • external storage block or stage (memory) – a number of typical devices associated with each "block" • i.e., keyboard or mouse with the input block – a flow of data through the computer under the direction of a program.	