CAPE COMPUTER SCIENCE UNIT ONE (1) - UNIT PLAN 2023-2024

UNIT TITLE: Problem Solving with Computers

GRADE/ LEVEL: Twelve (12)

DURATION: Four (4) Weeks

OVERVIEW: On completion of this module, students should: understand the problem-solving process; appreciate the role and importance of algorithms in the problem-solving process; understand the process of developing algorithms

SPECIFIC OBJECTIVES	SUB-TOPIC(S)	MAJOR CONCEPTS	TEACHING STRATEGIES	LEARNING ACTIVITIES	EVALUATION	RESOURCES	WK
Students should: **0. Understand the Requirements of the Syllabus and the Internal Assessment**	Syllabus Content Overview and delivery sequence IA Mark scheme Overview	• Module 2: Problem-Solving with Computers • Module 3: Programming • Module 1: Computer Organization • Definition of the Problems • Narrative and Flow Charts or Pseudocode • Coding of Program • Testing and presentation • Communication of Information				Computer, Presentation slides and syllabus.	**1**

1. Explain the concept of problem solving **2. Describe the stages in the problem-solving process**	Problem-solving process. Stages of the Problem solving process	Problem Solving definition • Problem definition. • Problem analysis. • Identify and evaluate possible solutions • Select and justify optimal solutions • Implementation and review.	✓ Cooperative learning ✓ Case Method	• Students will be given a step of the problem solving process in groups to explain.		Laptop, computer, slides, syllabus and textbook	**1**
3. Explain the concept of an algorithm. **4. Identify the necessary properties of algorithms that are well designed.** **5. Identify ways of representing algorithms**	Algorithms Properties of a well designed algorithm Ways of representing algorithms	Algorithm Definition. • A general solution to the problem • Clearly defined and unambiguous steps. • Finite number of steps. • Flow of control from one process to the next. Narrative, Flowchart & Pseudocode	✓ Cooperative learning ✓ Discussion	• Students will research the Properties of a well-designed algorithm for discussion. • Students will be given algorithms represented in the 3 ways for them to make comparisons, identifying some constructs are represented.	Verbal Recap	Laptop, computer, slides, syllabus and textbook	**1**

6. Explain constructs used in structured programming	Constructs used in structured programming	• Input and Output statements • Control Structures: o Sequence o Selection o Iteration or repetition (bounded and unbounded	✓ Cooperative learning ✓ Case Method	• Students will use control structures in the creation of algorithms in groups.	Students will review their algorithms with the aid of the teacher.	Laptop, computer, slides, syllabus and textbook	**2**
7. Interpret algorithms from case problems. **8. Correct algorithms from case problems.**	Tracing/Algorithm Review	Determination of output and correctness of a given algorithm Determine whether an algorithm achieves its stated objective and if not provide a correct algorithm	✓ Cooperative learning ✓ Case Method	• Students will be given algorithms to trace. • Students will analyse and fix incorrect algorithms.	Class review of trace tables Students will review each other's work.	Laptop, computer, slides, syllabus and textbook	**3**
9. Develop algorithms from case problems. **10. Explain the need for developing the logic of a computer program.**	Algorithm Development	Algorithms as logically sequenced instructions.	✓ Cooperative learning ✓ Case Method	• Discussion • Students will analyze the steps algorithms and determine correctness. • Students will explain the need for algorithms		Laptop, computer, slides, syllabus and textbook	**4**

UNIT TITLE: Programming

GRADE/ LEVEL: Twelve (12)

DURATION: Nine (9) Weeks

OVERVIEW: On completion of this module students should: appreciate the need for different programming languages and program translation; develop the ability to implement solution to problems using programming language.

SPECIFIC OBJECTIVES	SUB-TOPIC(S)	MAJOR CONCEPTS	TEACHING STRATEGIES	LEARNING ACTIVITIES	EVALUATION	RESOURCES	WK
Students should: **1. Identify the characteristics of different programming paradigms.**	Characteristics of different programming paradigms.	• Procedural or Imperative • Object-oriented • Functional • Declarative.	✓ Discussion ✓ Cooperative learning	Students will be grouped where they will be required to research characteristics of the different programming paradigms etc. for discussion.	Informal	Laptop, computer, slides, syllabus and textbook	**5**
2. Explain the need for different programming languages	Programming Languages	Appropriateness to applications (web application, games, formula translation, application for mobile devices)	✓ Discussion ✓ Cooperative learning	Students will explore content of the topic in both video and written.	Students will be required to write a short essay explaining the need for different programming languages.	Laptop, computer, slides, syllabus and textbook	**5**
SIX WEEK TEST							
3. Explain how Assemblers, Compilers and Virtual machines and interpreters are involved in the execution of High-level programming languages.	Program Translation	Stages in the translation process: • Lexical analysis • Syntax analysis • Semantic analysis • Intermediate code generation • Code optimization • Code generation.	✓ Discussion ✓ Cooperative learning	Students will discuss the different stages of the translation process.	Students will state the deliverables of each stage of the translation process.	Laptop, computer, slides, syllabus and textbook	**6**

	Programming Language Translators	Assemblers, Compiler Role of preprocessors; linkers.					
4. Assign variables to declared variables **5. Using input and output statements** **6. Develop good programming style**	Variable declaration and assignment Input and Output Statements Good programming practices	• Appropriate variable names. • Data types (Integer, float, double and char, char []) • Variable naming conventions Inputting data to variables, Output data from variables, print headings. Proper spacing (white spaces), indentation and comments	✓ Discussion ✓ Cooperative learning ✓ Demonstration	Students will declare proper variables to be used in their algorithms. Students will research and share amongst classmates how to input and output data to and from variables. Students will be required to add comments to their programs.		Laptop, computer, slides, syllabus and textbook	**7**
7. Choose appropriate conditional and iterative constructs **8. Use Conditional and Iterative Constructs**	Conditional and Iterative Constructs	Conditional • If Construct • If-then-Else Construct • Nested-If Constructs Iterative • While • Do-While • For	✓ Demonstration ✓ Cooperative Learning	Students will be given a chance to analyze a problem and decide which construct best suits the problem. Students will implement solutions to problems utilizing conditional and iterative		Laptop, computer, slides, syllabus and textbook	**7**

				constructs.			
9. Use arrays in programs	Arrays	• Read data into arrays • Output data from arrays	✓ Demonstration ✓ Cooperative Learning	Students will write simple C programs to perform	Students will test the correctness of their programs	Laptop, computer, slides, syllabus and textbook	8
		• Manipulate or modify data in arrays. • Character Arrays (String)	✓ Case Studies	basic functions with an array.	through the use of test data.		
10. Apply the techniques of structured decomposition to reorganize a program into smaller pieces	Functions	Write simple and clear functions;	✓ Cooperative Learning ✓ Case Studies	Students will work collaboratively to decompose given problems into functions.		Laptop, computer, slides, syllabus and textbook	**9**
11. Implement Algorithms to solve a given problem	Algorithm Implementation	• Write test and debug programs • Use of range tests and desk checks • Code debugging strategies (trace tables, watches)	✓ Demonstration ✓ Cooperative Learning	Students will be given erroneous programs/modules and asked to debug them.	Students will fix incorrect code given to them in a PowerPoint /Worksheet	Laptop, computer, slides, syllabus and textbook	**10**
12. Use records as a means of grouping related information.	Records	The concept of a Struct in C	✓ Demonstration ✓ Cooperative Learning	Students will work collaboratively to find various entities that can be represented as a struct.	Students will be given entities for which they will have to develop suitable structs.	Laptop, computer, slides, syllabus and textbook	**11**
13. Use text files to store data and records	Files	File Operations: Open, close, read, write, append.	✓ Demonstration ✓ Cooperative Learning	Students will work collaboratively after given adequate examples to add modules to perform file	Students will be required to prove correctness of their modules	Laptop, computer, slides, syllabus and textbook	**12 & 13**

				operations to their programs.	by testing them.		
SIX WEEK TEST							