

<b>Unit Title: Principles of Programming</b>	<b>Unit Code: PP</b>
<b>Level: 5</b>	<b>Learning Hours: 160</b>
<b>Learning Outcomes and Indicative Content:</b>	
Candidates will be able to:	
<ol style="list-style-type: none"> <li><b>1. Describe the normal steps in developing a program using a common high level programming language.</b> <ol style="list-style-type: none"> <li>1.1 Define requirements, analyse the problem, create an outline solution, code, test and document the software product.</li> <li>1.2 Compare the role of compilers and interpreters in the translation and execution of programs.</li> <li>1.3 Describe the main facilities provided in a typical Program Development Environment (PDE).</li> </ol> </li> <li><b>2. Declare and process appropriate data structures.</b> <ol style="list-style-type: none"> <li>2.1 Select suitable data types and identifiers for program variables and constants.</li> <li>2.2 Assign values and expressions to variables.</li> <li>2.3 Explain the precedence of arithmetic operators in expressions.</li> </ol> </li> <li><b>3. Apply selection constructs in conditional statements.</b> <ol style="list-style-type: none"> <li>3.1 Create simple conditional statements.</li> <li>3.2 Construct conditional statements containing Boolean expressions.</li> <li>3.3 Create nested conditional statements.</li> </ol> </li> <li><b>4. Employ repetition constructs in programs, including nested loops.</b> <ol style="list-style-type: none"> <li>4.1 Write correctly formed definite loops.</li> <li>4.2 Explain the need for indefinite loops and provide examples of their use.</li> <li>4.3 Distinguish between pre-conditioned and post-conditioned loops.</li> </ol> </li> <li><b>5. Declare and apply one-dimensional arrays.</b> <ol style="list-style-type: none"> <li>5.1 Declare one-dimensional arrays of suitable sizes and types.</li> <li>5.2 Process array elements using loops and subscripts.</li> <li>5.3 Construct applications such as simple sorting, searching and merging, using array structures.</li> </ol> </li> <li><b>6. Implement input and output statements.</b> <ol style="list-style-type: none"> <li>6.1 Input various data types through the keyboard for processing.</li> <li>6.2 Display and print results in a chosen format.</li> <li>6.3 Read data from a simple file and write to text or data files.</li> </ol> </li> </ol>	

- 7. Construct programs which employ procedures and functions.**
  - 7.1 Demonstrate the use of both simple procedures and procedures with parameters.
  - 7.2 Describe the use of functions built into the programming language.
  - 7.3 Show how to declare and employ user-defined functions.
  
- 8. Explain how to identify program errors and how to test programs in a systematic manner.**
  - 8.1 Name three types of program error and describe how they may be detected and corrected.
  - 8.2 Describe the stages of testing from unit module level to acceptance testing.
  - 8.3 Explain the importance of a test plan and of selecting appropriate values of test data.
  - 8.4 Construct a table to show how program variables change during the running of a process.
  
- 9. Complete the detailed documentation of one or more programs.**
  - 9.1 List the headings needed in a documentation folder.
  - 9.2 Assemble all the documentation components from the Introduction through to the Conclusion.
  - 9.3 Employ flowcharts, pseudocode or universal modelling language (UML) to record the logic of algorithms.
  
- 10. Demonstrate the ability to complete a programming project.**
  - 10.1 Receive and analyse a statement of program requirements for an application based on well-known teaching examples.
  - 10.2 Devise a suitable solution and proceed through recognised programming stages to the creation of a tested software product.
  - 10.3 Present the finished documentation to a recognised standard and within a specified period of time.
  
- 11. Demonstrate the skills required in the management of a software development project.**
  - 11.1 Recommend the most appropriate programming methodology and staffing structure for a software project.
  - 11.2 Prepare plans for the technical stages of a software project.
  - 11.3 Organise and supervise the work of junior staff, providing support where necessary.
  - 11.4 Establish standards for programming techniques.
  - 11.5 Ensure quality standards are achieved throughout the project.

**Assessment Criteria:**

- Assessment method: written examination
- Length of examination: three hours
- Candidates should answer four questions from a choice of eight, each question carrying equal marks.

**Recommended Reading**

Jarvis A, Blundell P, Reid M, *Software Development with Java* (2004),  
Heinemann  
ISBN: 0435471503

Jarvis A, Blundell P, Reid M, *Software Development with Visual Basic* (2004),  
Heinemann  
ISBN: 043547152X