

**NEW ZEALAND INSTITUTES OF TECHNOLOGY AND POLYTECHNIC
QUALIFICATIONS IN INFORMATION & COMMUNICATIONS TECHNOLOGY**

PRESCRIPTION: PP590 PROGRAMMING CONCEPTS AND TOOLS

AIM OF MODULE:	To enable students to design and write solutions to more complex programming problems that involve the use of a variety of logic methods and data sources.
CREDITS:	7
STUDENT LEARNING HOURS:	70
ASSUMED KNOWLEDGE	PP490
CONTENT REVISED:	2002 (new)
PRESCRIPTION EXPIRY DATE:	November 2013
NOTE:	THIS IS A COMPULSORY DipICT L5 MODULE Students may omit this compulsory module if they have achieved passes in both PD500 and PP400.

Level and Assessment Schedule

TOPICS	Highest Skill Level				Suggested Assessment Percentage
	R	C	A	P	
1. Program development cycle		*			5
2. Problem decomposition and logic depiction methods			*		10
3. Design of logic solutions using sequences, selections and repetitions and a variety of input / output media			*		20
4. Desk checks			*		10
5. Translate appropriate logic solutions into selected programming language			*		40
6. Documentation			*		5
7. Testing			*		5
8. Languages		*			5
					<hr/> 100 <hr/>

LEARNING OUTCOMES

The student will:

- C 1 Describe the program development cycle.
- A 2 Apply techniques of problem decomposition and apply a variety of logic depiction methods to illustrate solutions to common types of programming tasks.
- A 3 Demonstrate appropriate use of sequence, selection and iteration to design well structured procedures for solving programming tasks that may involve a variety of data sources.
- A 4 Desk check the methodology using test data.
- A 5 Use appropriate programming features to correctly translate a program solution from an appropriate depiction of logic into the selected programming language. Compile, resolve syntax errors, test the programs, and modify the programs as required to meet the original design objectives.
- A 6 Document programs internally and externally according to given standards.
- A 7 Use valid program testing techniques to ensure programmes meet their design objectives.
- C 8 Describe the purpose and characteristics of computer programming languages from first generation to current languages including reasons for their development.

CONTENT

NOTE

- It is expected that a depiction method related to the programming language taught in this module will be covered to a greater depth.
- 1 The program development cycle stages are to be explained. The activities that are taught in lectures should be related to the cycle steps.
- 2 Suitable logic depiction methods such as structure diagrams, structured English, and UML may be used.
- Programs need not relate to business data processing but the selection of tasks should introduce design techniques that will be required for typical data processing problems.
- A *maximum* of three levels of nesting of selection / iteration is recommended.

- Examples of procedures to indicate the level of difficulty:
 - Calculate tax on a graded scale from hours and unit pay
 - Calculation of sum and average, output of highest and lowest of a list of numbers ended by "rogue" values
 - Reading data in text format from a file, selecting records by specific criteria from input, simple calculations on selected data and writing to an output file.

- It is envisaged that the principal logic solution method taught in PP490 will be expanded upon and that additional methods will also be introduced.

- 3** Problem solution is now more complex and should be based on procedures, functions and sub-programs all controlled by main line logic.

- 4** Programming features such as input, output, operations (eg. arithmetic) on data, data storage including files should be included.

- The action of individual items of syntax in the selected programming language should be explained. This includes information about any data elements it works with, produces or changes.

- The subset will include logic control, assignment and comparison operators that are required to write programs for this module.

- Examples of suitable languages:
 - Visual Basic
 - Pascal
 - Java
 - Jade

- An integrated programming environment using either a procedural or object oriented language is recommended.

- 5** The student must write the program from the selected logic depiction method and ensure that all original design objectives are met.

- 6** An appropriate documentation standard should be introduced to ensure students learn the importance of retaining original programme documentation as well as managing changes to documentation. Documentation standards are to include both internal source code programme documentation and paper based documentation. The documentation standard should include such references as author, purpose of program, date written, modifications, variable names, procedure names, logic plan, testing etc.

- 7 A method (or methods) of progressively testing a program to ensure that it meets all design objectives (including all possible processing options) is to be introduced. This will include the techniques of selecting test data values and the ability to depict the results the programme is to produce using these test data values.
- 8 The development of languages from first generation to present day should be covered. A comparison of the attributes of the various generations of languages should be covered.

NOTES AND COMMENTS

- The intention of this module is to get the student to implement complex problem solutions they have completed in the logic section of this module, not to produce commercial programs.
- The module requires students to acquire skills in taking the fundamental building blocks of procedure design (selection, iteration and sequence) and applying these to simple problems. The vehicle for testing the resulting procedures will be the selected programming language. Equal assessment is to be given to both program design and the ability to translate that design into the selected language.
- The training provider should ensure that the appropriate mix of theory and practical application is maintained throughout the teaching of this module. Theory should be re-enforced with practical, **not** treated as a separate topic.