



NATIONAL CERTIFICATES (VOCATIONAL)

ASSESSMENT GUIDELINES

PRINCIPLES OF COMPUTER PROGRAMMING NQF Level 3

September 2007

PRINCIPLES OF COMPUTER PROGRAMMING– LEVEL 3

CONTENTS

SECTION A: PURPOSE OF THE SUBJECT ASSESSMENT GUIDELINES

SECTION B: ASSESSMENT IN THE NATIONAL CERTIFICATES (VOCATIONAL)

- 1 Assessment in the National Certificates (Vocational)**
- 2 Assessment framework for vocational qualifications**
 - 2.1 Internal continuous assessment (ICASS)
 - 2.2 External summative assessment (ESASS)
- 3 Moderation of assessment**
 - 3.1 Internal moderation
 - 3.2 External moderation
- 4 Period of validity of internal continuous assessment (ICASS)**
- 5 Assessor requirements**
- 6 Types of assessment**
 - 6.1 Baseline assessment
 - 6.2 Diagnostic assessment
 - 6.3 Formative assessment
 - 6.4 Summative assessment
- 7 Planning assessment**
 - 7.1 Collecting evidence
 - 7.2 Recording
 - 7.3 Reporting
- 8 Methods of assessment**
- 9 Instruments and tools for collecting evidence**
- 10 Tools for assessing student performance**
- 11 Selecting and/or designing recording and reporting systems**
- 12 Competence descriptions**
- 13 Strategies for collecting evidence**
 - 13.1 Record sheets
 - 13.2 Checklists

SECTION C: ASSESSMENT IN PRINCIPLES OF COMPUTER PROGRAMMING

- 1 Schedule of assessment**
- 2 Recording and reporting**
- 3 Internal assessment of Subject Outcomes in Principles of Computer Programming - Level 3**
- 4 Specifications for external assessment in Principles of Computer Programming Level 3**
 - 4.1 Integrated summative assessment task (ISAT)
 - 4.2 National examination

SECTION A: PURPOSE OF THE SUBJECT ASSESSMENT GUIDELINES

This document provides the lecturer with guidelines to develop and implement a coherent, integrated assessment system for Introduction to Systems Development in the National Certificates (Vocational). It must be read with the *National Policy Regarding Further Education and Training Programmes: Approval of the Documents, Policy for the National Certificates (Vocational) Qualifications at Levels 2 to 4 on the National Qualifications Framework (NQF)*. This assessment guideline will be used for National Qualifications Framework Levels 2-4.

This document explains the requirements for the internal and external subject assessment. The lecturer must use this document with the *Subject Guidelines: Introduction to Systems Development* to prepare for and deliver Introduction to Systems Development. Lecturers should use a variety of resources and apply a range of assessment skills in the setting, marking and recording of assessment tasks.

SECTION B: ASSESSMENT IN THE NATIONAL CERTIFICATES (VOCATIONAL)

1 ASSESSMENT IN THE NATIONAL CERTIFICATES (VOCATIONAL)

Assessment in the National Certificates (Vocational) is underpinned by the objectives of the National Qualifications Framework (NQF). These objectives are to:

- Create an integrated national framework for learning achievements.
- Facilitate access to and progression within education, training and career paths.
- Enhance the quality of education and training.
- Redress unfair discrimination and past imbalances and thereby accelerate employment opportunities.
- Contribute to the holistic development of the student by addressing:
 - social adjustment and responsibility;
 - moral accountability and ethical work orientation;
 - economic participation; and
 - nation-building.

The principles that drive these objectives are:

- **Integration**

To adopt a unified approach to education and training that will strengthen the human resources development capacity of the nation.

- **Relevance**

To be dynamic and responsive to national development needs.

- **Credibility**

To demonstrate national and international value and recognition of qualification and acquired competencies and skills.

- **Coherence**

To work within a consistent framework of principles and certification.

- **Flexibility**

To allow for creativity and resourcefulness when achieving Learning Outcomes, to cater for different learning styles and use a range of assessment methods, instruments and techniques.

- **Participation**

To enable stakeholders to participate in setting standards and co-ordinating the achievement of the qualification.

- **Access**

To address barriers to learning at each level to facilitate students' progress.

- **Progression**

To ensure that the qualification framework permits individuals to move through the levels of the national qualification via different, appropriate combinations of the components of the delivery system.

- **Portability**

To enable students to transfer credits of qualifications from one learning institution and/or employer to another institution or employer.

- **Articulation**

To allow for vertical and horizontal mobility in the education system when accredited pre-requisites have been successfully completed.

- **Recognition of Prior Learning**

To grant credits for a unit of learning following an assessment or if a student possesses the capabilities specified in the outcomes statement.

- **Validity of assessments**

To ensure assessment covers a broad range of knowledge, skills, values and attitudes (SKVAs) needed to demonstrate applied competency. This is achieved through:

- clearly stating the outcome to be assessed;
- selecting the appropriate or suitable evidence;
- matching the evidence with a compatible or appropriate method of assessment; and
- selecting and constructing an instrument(s) of assessment.

- **Reliability**

To assure assessment practices are consistent so that the same result or judgment is arrived at if the assessment is replicated in the same context. This demands consistency in the interpretation of evidence; therefore, careful monitoring of assessment is vital.

- **Fairness and transparency**

To verify that no assessment process or method(s) hinders or unfairly advantages any student. The following could constitute unfairness in assessment:

- Inequality of opportunities, resources or teaching and learning approaches
- Bias based on ethnicity, race, gender, age, disability or social class
- Lack of clarity regarding Learning Outcome being assessed
- Comparison of students' work with other students, based on learning styles and language

- **Practicability and cost-effectiveness**

To integrate assessment practices within an outcomes-based education and training system and strive for cost and time-effective assessment.

2 ASSESSMENT FRAMEWORK FOR VOCATIONAL QUALIFICATIONS

The assessment structure for the National Certificates (Vocational) qualification is as follows:

2.1 Internal continuous assessment (ICASS)

Knowledge, skills values, and attitudes (SKVAs) are assessed throughout the year using assessment instruments such as projects, tests, assignments, investigations, role-play and case studies. The internal continuous assessment (ICASS) practical component is undertaken in a real workplace, a workshop or a "Structured Environment". This component is moderated internally and externally quality assured by Umalusi. All internal continuous assessment (ICASS) evidence is kept in a Portfolio of Evidence (PoE) and must be readily available for monitoring, moderation and verification purposes.

2.2 External summative assessment (ESASS)

The external summative assessment is either a single or a set of written papers set to the requirements of the Subject Learning Outcomes. The Department of Education administers the theoretical component according to relevant assessment policies.

A compulsory component of external summative assessment (ESASS) is the **integrated summative assessment task (ISAT)**. This assessment task draws on the students' cumulative learning throughout the year. The task requires **integrated application of competence** and is executed under strict assessment conditions. The task should take place in a simulated or "Structured Environment". The integrated summative assessment task (ISAT) is the most significant test of students' ability to apply their acquired knowledge.

The integrated assessment approach allows students to be assessed in more than one subject with the same integrated summative assessment task (ISAT).

External summative assessments will be conducted annually between October and December, with provision made for supplementary sittings.

3 MODERATION OF ASSESSMENT

3.1 Internal moderation

Assessment must be moderated according to the internal moderation policy of the Further Education and Training (FET) college. Internal college moderation is a continuous process. The moderator's involvement starts with the planning of assessment methods and instruments and follows with continuous collaboration with and support to the assessors. Internal moderation creates common understanding of Assessment Standards and maintains these across vocational programmes.

3.2 External moderation

External moderation is conducted by the Department of Education, Umalusi and, where relevant, an Education and Training Quality Assurance (ETQA) body according to South African Qualifications Authority (SAQA) and Umalusi standards and requirements.

The external moderator:

- monitors and evaluates the standard of all summative assessments;
- maintains standards by exercising appropriate influence and control over assessors;
- ensures proper procedures are followed;
- ensures summative integrated assessments are correctly administered;
- observes a minimum sample of ten (10) to twenty-five (25) percent of summative assessments;
- gives written feedback to the relevant quality assessor; and
- moderates in case of a dispute between an assessor and a student.

Policy on inclusive education requires that assessment procedures for students who experience barriers to learning be customised and supported to enable these students to achieve their maximum potential.

4 PERIOD OF VALIDITY OF INTERNAL CONTINUOUS ASSESSMENT (ICASS)

The period of validity of the internal continuous assessment mark is determined by the *National Policy on the Conduct, Administration and Management of the Assessment of the National Certificates (Vocational)*.

The internal continuous assessment (ICASS) must be re-submitted with each examination enrolment for which it constitutes a component.

5 ASSESSOR REQUIREMENTS

Assessors must be subject specialists and should ideally be declared competent against the standards set by the ETDP SETA. If the lecturer conducting the assessments has not been declared a competent assessor, an assessor who has been declared competent may be appointed to oversee the assessment process to ensure the quality and integrity of assessments.

6 TYPES OF ASSESSMENT

Assessment benefits the student and the lecturer. It informs students about their progress and helps lecturers make informed decisions at different stages of the learning process. Depending on the intended purpose, different types of assessment can be used.

6.1 Baseline assessment

At the beginning of a level or learning experience, baseline assessment establishes the knowledge, skills, values and attitudes (SKVAs) that students bring to the classroom. This knowledge assists lecturers to plan learning programmes and learning activities.

6.2 Diagnostic assessment

This assessment diagnoses the nature and causes of learning barriers experienced by specific students. It is followed by guidance, appropriate support and intervention strategies. This type of assessment is useful to make referrals for students requiring specialist help.

6.3 Formative assessment

This assessment monitors and supports teaching and learning. It determines student strengths and weaknesses and provides feedback on progress. It determines if a student is ready for summative assessment.

6.4 Summative assessment

This type of assessment gives an overall picture of student progress at a given time. It determines whether the student is sufficiently competent to progress to the next level.

7 PLANNING ASSESSMENT

An assessment plan should cover three main processes:

7.1 Collecting evidence

The assessment plan indicates which Subject Outcomes and Assessment Standards will be assessed, what assessment method or activity will be used and when this assessment will be conducted.

7.2 Recording

Recording refers to the assessment instruments or tools with which the assessment will be captured or recorded. Therefore, appropriate assessment instruments must be developed or adapted.

7.3 Reporting

All the evidence is put together in a report to deliver a decision for the subject.

8 METHODS OF ASSESSMENT

Methods of assessment refer to who carries out the assessment and includes lecturer assessment, self-assessment, peer assessment and group assessment.

LECTURER ASSESSMENT	The lecturer assesses students' performance against given criteria in different contexts, such as individual work, group work, etc.
SELF-ASSESSMENT	Students assess their own performance against given criteria in different contexts, such as individual work, group work, etc.
PEER ASSESSMENT	Students assess another student's or group of students' performance against given criteria in different contexts, such as individual work, group work, etc.
GROUP ASSESSMENT	Students assess the individual performance of other students within a group or the overall performance of a group of students against given criteria.

9 INSTRUMENTS AND TOOLS FOR COLLECTING EVIDENCE

All evidence collected for assessment purposes is kept or recorded in the student's Portfolio of Evidence (PoE).

The following table summarises a variety of methods and instruments for collecting evidence. A method and instrument is chosen to give students ample opportunity to demonstrate the Subject Outcome has been attained. This will only be possible if the chosen methods and instruments are appropriate for the target group and the Specific Outcome being assessed.

	METHODS FOR COLLECTING EVIDENCE		
	Observation-based (Less structured)	Task-based (Structured)	Test-based (More structured)
Assessment instruments	<ul style="list-style-type: none"> • Observation • Class questions • Lecturer, student, parent discussions 	<ul style="list-style-type: none"> • Assignments or tasks • Projects • Investigations or research • Case studies • Practical exercises • Demonstrations • Role-play • Interviews 	<ul style="list-style-type: none"> • Examinations • Class tests • Practical examinations • Oral tests • Open-book tests
Assessment tools	<ul style="list-style-type: none"> • Observation sheets • Lecturer's notes • Comments 	<ul style="list-style-type: none"> • Checklists • Rating scales • Rubrics 	<ul style="list-style-type: none"> • Marks (e.g. %) • Rating scales (1-7)
Evidence	<ul style="list-style-type: none"> • Focus on individual students • Subjective evidence based on lecturer observations and impressions 	<p>Open middle: Students produce the same evidence but in different ways.</p> <p>Open end: Students use same process to achieve different results.</p>	Students answer the same questions in the same way, within the same time.

10 TOOLS FOR ASSESSING STUDENT PERFORMANCE

Rating scales are marking systems where a symbol (such as 1 to 7) or a mark (such as 5/10 or 50%) is defined in detail. The detail is as important as the coded score. Traditional marking, assessment and evaluation mostly used rating scales without details such as what was right or wrong, weak or strong, etc.

Task lists and **checklists** show the student what needs to be done. These consist of short statements describing the expected performance in a particular task. The statements on the checklist can be ticked off when the student has adequately achieved the criterion. Checklists and task lists are useful in peer or group assessment activities.

Rubrics are a hierarchy (graded levels) of criteria with benchmarks that describe the minimum level of acceptable performance or achievement for each criterion. Using rubrics is a different way of assessing and cannot be compared to tests. Each criterion described in the rubric must be assessed separately. Mainly two types of rubrics, namely holistic and analytical, are used.

11 SELECTING AND/OR DESIGNING RECORDING AND REPORTING SYSTEMS

The selection or design of recording and reporting systems depends on the purpose of recording and reporting student achievement. **Why** particular information is recorded and **how** it is recorded determine which instrument will be used.

Computer-based systems, for example spreadsheets, are cost and time effective. The recording system should be user-friendly and information should be easily accessed and retrieved.

12 COMPETENCE DESCRIPTIONS

All assessment should award marks to evaluate specific assessment tasks. However, marks should be awarded against rubrics and not be simply a total of ticks for right answers. Rubrics should explain the competence level descriptors for the skills, knowledge, values and attitudes (SKVAs) that a student must demonstrate to achieve each level of the rating scale.

When lecturers or assessors prepare an assessment task or question, they must ensure that the task or question addresses an aspect of a Subject Outcome. The relevant Assessment Standard must be used to create the rubric to assess the task or question. The descriptions must clearly indicate the minimum level of attainment for each category on the rating scale.

13 STRATEGIES FOR COLLECTING EVIDENCE

A number of different assessment instruments may be used to collect and record evidence. Examples of instruments that can be (adapted and) used in the classroom include:

13.1 Record sheets

The lecturer observes students working in a group. These observations are recorded in a summary table at the end of each project. The lecturer can design a record sheet to observe students' interactive and problem-solving skills, attitudes towards group work and involvement in a group activity.

13.2 Checklists

Checklists should have clear categories to ensure that the objectives are effectively met. The categories should describe how the activities are evaluated and against what criteria they are evaluated. Space for comments is essential.

SECTION C: ASSESSMENT IN PRINCIPLES OF COMPUTER PROGRAMMING

1 SCHEDULE OF ASSESSMENT

At NQF levels 2, 3 and 4, lecturers will conduct assessments as well as develop a schedule of formal assessments that will be undertaken in the year. All three levels also have an external examination that accounts for 50 percent of the total mark. The marks allocated to assessment tasks completed during the year, kept or recorded in a Portfolio of Evidence (PoE), account for the other 50 percent.

The Portfolio of Evidence (PoE) and the external assessment include practical and written components. The practical assessment in Principles of Computer Programming must, where necessary, be subjected to external moderation by Umalusi or an appropriate Education and Training Quality Assurance (ETQA) body, appointed by the Umalusi Council in terms of Section 28(2) of the *General and Further Education and Training Quality Assurance Act, 2001 (Act No. 58 of 2001)*.

2 RECORDING AND REPORTING

Introduction to Systems Development, as is the case for all the other Vocational subjects, is assessed according to five levels of competence. The level descriptions are explained in the following table.

Scale of Achievement for the Vocational component

RATING CODE	RATING	MARKS %
5	Outstanding	80-100
4	Highly competent	70-79
3	Competent	50-69
2	Not yet competent	40-49
1	Not achieved	0-39

The programme of assessment should be recorded in the Lecturer's Portfolio of Assessment for each subject. The following at least should be included in the Lecturer's Assessment Portfolio:

- A contents page
- The formal schedule of assessment
- The requirements for each assessment task
- The tools used for each assessment task
- Recording instrument(s) for each assessment task
- A mark sheet and report for each assessment task

The college must standardise these documents.

The student's Portfolio of Evidence (PoE) must include at least:

- A contents page
- The assessment tasks according to the assessment schedule

- The assessment tools or instruments for the task
- A record of the marks (and comments) achieved for each task

Where a task cannot be contained as evidence in the Portfolio of Evidence (PoE), its exact location must be recorded and it must be readily available for moderation purposes.

**ASSESSMENT OF PRINCIPLES OF COMPUTER
PROGRAMMING
LEVEL 3**

3 INTERNAL ASSESSMENT OF SUBJECT OUTCOMES IN PRINCIPLES OF COMPUTER PROGRAMMING - LEVEL 3

Topic 1: Principles of electronic logic for computing

SUBJECT OUTCOME	
1.1 Perform Boolean logic algebra operations.	
ASSESSMENT STANDARD	LEARNING OUTCOME
<ul style="list-style-type: none"> The performance demonstrates algebraic laws by using Boolean algebra. <i>Range: commutative; associative; distributive; tautology</i> 	<ul style="list-style-type: none"> Demonstrate algebraic laws by using Boolean algebra.
<ul style="list-style-type: none"> The performance explains and translates Boolean expressions to logic diagrams and vice versa. 	<ul style="list-style-type: none"> Explain and demonstrate how to translate Boolean expressions to logic diagrams and vice versa.
<ul style="list-style-type: none"> The performance explains and translates truth tables using Boolean expressions. 	<ul style="list-style-type: none"> Explain and demonstrate how to translate truth tables using Boolean expressions.
<ul style="list-style-type: none"> The performance simplifies Boolean expressions by using decision tables. 	<ul style="list-style-type: none"> Simplify Boolean expressions by using decision tables.
<ul style="list-style-type: none"> The performance describes the output of a truth table and logic diagram for a given series of inputs. 	<ul style="list-style-type: none"> Explain the output of a truth table and logic diagram for a given series of inputs.
<ul style="list-style-type: none"> The performance describes a logic diagram for a simple truth table and simple written statement. 	<ul style="list-style-type: none"> Describe a logic diagram for a simple truth table and simple written statement.
ASSESSMENT TASKS OR ACTIVITIES	
<ul style="list-style-type: none"> Class tests Projects/practical work Assignments Group work 	

SUBJECT OUTCOME	
1.2 Demonstrate knowledge of logic gates	
ASSESSMENT STANDARD	LEARNING OUTCOME
<ul style="list-style-type: none"> The demonstration states the symbol, truth table, logic diagram and Boolean expression for logic gates. <i>Range: AND; OR; NOT; NAND; NOR</i> 	<ul style="list-style-type: none"> State the symbol, truth table, logic diagram and Boolean expression for logic gates.
<ul style="list-style-type: none"> The demonstration combines logic gates to perform the functions of the XOR gate. 	<ul style="list-style-type: none"> Combine logic gates to perform the functions of the XOR gate.
<ul style="list-style-type: none"> The demonstration identifies integrated circuits that implement gates for the computer systems. 	<ul style="list-style-type: none"> Identify the integrated circuits that implement gates for the computer systems.
<ul style="list-style-type: none"> The demonstration explains memory chips in terms of their use, functions, and operational characteristics. 	<ul style="list-style-type: none"> Explain the memory chips in terms of their use, functions, and operational characteristics.
ASSESSMENT TASKS OR ACTIVITIES	
<ul style="list-style-type: none"> Class tests Projects/practical work Assignments Group work 	

SUBJECT OUTCOME	
1.3 Convert numbers using decoders and encoders.	
ASSESSMENT STANDARD	LEARNING OUTCOME
<ul style="list-style-type: none"> Conversions of numbers between combinations of decimal, binary and hexadecimal are demonstrated, using encoders. 	<ul style="list-style-type: none"> Demonstrate how to convert numbers between the combinations of decimal, binary and hexadecimal, using encoders.
<ul style="list-style-type: none"> Conversions of numbers between combinations of decimal, binary and hexadecimal are demonstrated, using decoders. 	<ul style="list-style-type: none"> Demonstrate how to convert numbers between combinations of decimal, binary and hexadecimal, using decoders.
<ul style="list-style-type: none"> Binary numbers are added and subtracted. 	<ul style="list-style-type: none"> Add and subtract binary numbers.
<ul style="list-style-type: none"> Two's complement is applied as a data representation in the subtraction of binary numbers. 	<ul style="list-style-type: none"> Apply two's complement as a data representation in the subtraction of binary numbers.
ASSESSMENT TASKS OR ACTIVITIES	
<ul style="list-style-type: none"> Class tests Projects/Practical work Assignments Group work 	

SUBJECT OUTCOME	
1.4 Demonstrate knowledge of the general principles of logic devices.	
ASSESSMENT STANDARD	LEARNING OUTCOME
<ul style="list-style-type: none"> The demonstration explains and draws symbols for logic devices according to industry conventions. <i>Range: multivibrators; bistables; Shift registers; Counters</i> 	<ul style="list-style-type: none"> Explain and draw symbols for logic devices according to industry conventions.
<ul style="list-style-type: none"> The demonstration identifies operations by describing different logic devices. 	<ul style="list-style-type: none"> Identify operations by describing different logic devices.
ASSESSMENT TASKS OR ACTIVITIES	
<ul style="list-style-type: none"> Class tests Projects/Practical work Assignments Group work 	

SUBJECT OUTCOMES	
1.5 Demonstrate knowledge of integrated circuit specifications.	
ASSESSMENT STANDARD	LEARNING OUTCOME
<ul style="list-style-type: none"> The demonstration identifies and explains the instruction sets for the integrated circuit. 	<ul style="list-style-type: none"> Explain and identify the instruction sets for the integrated circuit.
<ul style="list-style-type: none"> The demonstration identifies and explains the method that a CPU uses for addressing of instructions and the method for addressing of data. <i>Range: CISC and RISC</i> 	<ul style="list-style-type: none"> Explain and identify the method that a CPU uses for addressing of instructions and the method for addressing data.
ASSESSMENT TASKS OR ACTIVITIES	
<ul style="list-style-type: none"> Class tests Projects/Practical work Assignments Group work 	

Topic 2: Program development environments

SUBJECT OUTCOMES	
2.1 Demonstrate an understanding of different data representations used in computer programs.	
ASSESSMENT STANDARD	LEARNING OUTCOME
<ul style="list-style-type: none"> The demonstration applies different number conversion techniques between data types (at least two). <i>Range: type casting, explicit and implicit</i> 	<ul style="list-style-type: none"> Explain and apply different number conversion techniques between data types.
<ul style="list-style-type: none"> The demonstration explains the purpose a logical data type. <i>Range: flags, conditions resulting in True/False (AND OR NOT)</i> 	<ul style="list-style-type: none"> Explain the purpose of a logical data type.
<ul style="list-style-type: none"> The demonstration differentiates between and explains different internal representations of data types. <i>Range: signed numbers, floating point numbers, characters, arrays, pointers</i> 	<ul style="list-style-type: none"> Explain and differentiate between different internal representations of data types.
ASSESSMENT TASKS OR ACTIVITIES	
<ul style="list-style-type: none"> Class tests Projects/Practical work Assignments Group work 	

SUBJECT OUTCOMES	
2.2 Demonstrate an understanding of high level programming language concepts	
ASSESSMENT STANDARD	LEARNING OUTCOME
<ul style="list-style-type: none"> The demonstration explains constants and variables. 	<ul style="list-style-type: none"> Explain constants and variables.
<ul style="list-style-type: none"> The demonstration explains and illustrates the concepts of operators and expressions. <i>Range: arithmetic operators (+; -; *; /), logical operators (=; <; >), execution sequence, operator precedence</i> 	<ul style="list-style-type: none"> Explain and illustrate the concepts of operators and expressions.
<ul style="list-style-type: none"> The demonstration explains and illustrates different modular programming features and parameter passing. <i>Range: user defined functions and procedures,</i> 	<ul style="list-style-type: none"> Explain and illustrate different modular programming features and parameter passing.
ASSESSMENT TASKS OR ACTIVITIES	
<ul style="list-style-type: none"> Class tests Projects/Practical work Assignments Group work 	

SUBJECT OUTCOMES	
2.3 Explain and use visual programming language concepts	
ASSESSMENT STANDARD	LEARNING OUTCOME
<ul style="list-style-type: none"> The demonstration explains rapid application development Global and Local variables 	<ul style="list-style-type: none"> Explain the concept of rapid application development.
<ul style="list-style-type: none"> The demonstration explains event driven programming 	<ul style="list-style-type: none"> Explain event driven programming.
<ul style="list-style-type: none"> Visual programming language is explained and used 	<ul style="list-style-type: none"> Explain and use visual programming language.
<ul style="list-style-type: none"> The demonstration explains and illustrates rapid application development. 	<ul style="list-style-type: none"> Explain and illustrate rapid application development.
<ul style="list-style-type: none"> Event driven programming is explained and implemented. 	<ul style="list-style-type: none"> Explain and implement event driven programming.
ASSESSMENT TASKS OR ACTIVITIES	
<ul style="list-style-type: none"> Class tests Projects/Practical work Assignments Group work 	

Topic 3: Data Structures

SUBJECT OUTCOME	
3.1 Demonstrate an understanding of computer data structures	
ASSESSMENT STANDARD	LEARNING OUTCOME
<ul style="list-style-type: none"> The demonstration identifies and describes data structures. 	<ul style="list-style-type: none"> Identify and describe the concept data structures.
<ul style="list-style-type: none"> The demonstration identifies different data structure types. <i>Range: abstract data types , arrays, text files, data bases</i> 	<ul style="list-style-type: none"> Identify different data structure types.
<ul style="list-style-type: none"> The demonstration explains and illustrates features of different data structures. <i>Range: arrays, text files, databases</i> 	<ul style="list-style-type: none"> Explain and illustrate features of the different data structures.
<ul style="list-style-type: none"> The demonstration differentiates between types and uses of computer data structures. <i>Range: abstract data types ,arrays, text files, databases</i> 	<ul style="list-style-type: none"> Differentiate between types and uses of data structures.
<ul style="list-style-type: none"> The use of data structures when demonstrating simple searching techniques is explained. <i>Range: sequential search, binary search</i> 	<ul style="list-style-type: none"> Explain how to use data structures when demonstrating simple searching techniques.
<ul style="list-style-type: none"> The use of data structures when demonstrating simple sorting techniques is explained. <i>Range: selection, bubble</i> 	<ul style="list-style-type: none"> Explain how to use data structures when demonstrating simple sorting techniques.
<ul style="list-style-type: none"> The use of data structures when demonstrating manipulation of data is explained. <i>Range: inserting, deleting, modifying</i> 	<ul style="list-style-type: none"> Explain how to use data structures when demonstrating the manipulation of data.
ASSESSMENT TASKS OR ACTIVITIES	
<ul style="list-style-type: none"> Class tests Projects/Practical work Assignments Group work 	

Topic 4: Program design

SUBJECT OUTCOME	
4.1 Analyse the given problem.	
ASSESSMENT STANDARD	LEARNING OUTCOME
<ul style="list-style-type: none"> The given problem is described. 	<ul style="list-style-type: none"> Describe the given problem.
<ul style="list-style-type: none"> The program requirements are identified. <i>Range: design specifications, IPO, data requirements</i> 	<ul style="list-style-type: none"> Identify the program requirements.
ASSESSMENT TASKS OR ACTIVITIES	
<ul style="list-style-type: none"> Class tests Projects/Practical work Assignments Group work 	

SUBJECT OUTCOME	
4.2 Design and code a computer program to solve a given problem.	
ASSESSMENT STANDARD	LEARNING OUTCOME
<ul style="list-style-type: none"> An appropriate programming language is identified to solve a given problem. 	<ul style="list-style-type: none"> Identify an appropriate programming language to solve a given problem.
<ul style="list-style-type: none"> The appropriate techniques and data structures are implemented to solve the given problem. 	<ul style="list-style-type: none"> Implement the appropriate techniques and data structures to solve the given problem.
<ul style="list-style-type: none"> An algorithm is developed for the given problem. 	<ul style="list-style-type: none"> Develop an algorithm for the given problem.
<ul style="list-style-type: none"> The appropriate tools, techniques and data structures are implemented when coding the solution to the problem. <i>Range: sequential, selection structures, iteration structures and appropriate techniques such as searching and sorting and data manipulation techniques</i> 	<ul style="list-style-type: none"> Implement appropriate tools, techniques and data structures in coding the solution to the problem.
<ul style="list-style-type: none"> The solution to the problem is coded. 	<ul style="list-style-type: none"> Code the solution to the problem.
<ul style="list-style-type: none"> The solution to the problem is tested and debugged. 	<ul style="list-style-type: none"> Test and debug the solution to the problem.
ASSESSMENT TASKS OR ACTIVITIES	
<ul style="list-style-type: none"> Class tests Projects/Practical work Assignments Group work 	

Topic 5: Database application development

SUBJECT OUTCOME	
5.1 Plan and design a database to provide a solution to a given problem	
ASSESSMENT STANDARD	LEARNING OUTCOME
<ul style="list-style-type: none"> A working plan is developed to meet the requirements of a supplied brief. 	<ul style="list-style-type: none"> Develop a working plan to meet the requirements of a supplied brief.
<ul style="list-style-type: none"> The plan identifies the purpose of the database. 	<ul style="list-style-type: none"> Explain the purpose of the database.
<ul style="list-style-type: none"> The plan includes a basic outline of the database specifications and/or features required to provide a solution. 	<ul style="list-style-type: none"> Outline the database specifications and/or features required to provide a solution.

ASSESSMENT TASKS OR ACTIVITIES
<ul style="list-style-type: none"> • Class tests • Projects/Practical work • Assignments • Group work

SUBJECT OUTCOME	
5.2 Create a database according to the design using a database package	
ASSESSMENT STANDARD	LEARNING OUTCOME
<ul style="list-style-type: none"> • Appropriate tables are created 	<ul style="list-style-type: none"> • Create appropriate tables.
<ul style="list-style-type: none"> • Appropriate keys are set up <i>Range: Primary Keys, foreign Keys, alternate keys, indices</i> 	<ul style="list-style-type: none"> • Set up appropriate keys.
<ul style="list-style-type: none"> • A relationship between two tables is set up <i>Range: 1:1 , 1:Many</i> 	<ul style="list-style-type: none"> • Set up a relationship between two tables.
ASSESSMENT TASKS OR ACTIVITIES	
<ul style="list-style-type: none"> • Class tests • Projects/Practical work • Assignments • Group work 	

SUBJECT OUTCOME	
5.3 Create database forms.	
ASSESSMENT STANDARD	LEARNING OUTCOME
<ul style="list-style-type: none"> • Applicable forms are created for data input and manipulation. <i>Range: Simple forms, master-detail, sub-forms</i> 	<ul style="list-style-type: none"> • Create applicable forms for data input and manipulation.
<ul style="list-style-type: none"> • The form design is modified <i>Range: header, footer, layout, fields, components</i> 	<ul style="list-style-type: none"> • Modify the form design <i>Range: header, footer, layout, fields, components</i>
<ul style="list-style-type: none"> • Data is captured into the relevant tables using the developed form. 	<ul style="list-style-type: none"> • Capture data into the relevant tables using the developed form.
ASSESSMENT TASKS OR ACTIVITIES	
<ul style="list-style-type: none"> • Class tests • Projects/Practical work • Assignments • Group work 	

SUBJECT OUTCOME	
5.4 Use different methods to create different queries.	
ASSESSMENT STANDARD	LEARNING OUTCOME
<ul style="list-style-type: none"> A query is created with a query wizard. 	<ul style="list-style-type: none"> Create a query with a query wizard.
<ul style="list-style-type: none"> A query is created using simple SQL statements written by the developer. <i>Range: (SELECT, FROM WHERE, with single joins) HAVING</i> <i>Group functions: SUM, MIN, MAX, AVG, COUNT, ORDER BY, GROUP BY</i> 	<ul style="list-style-type: none"> Create a query using simple SQL statements written by the developer
<ul style="list-style-type: none"> The query is saved 	<ul style="list-style-type: none"> Save the query
<ul style="list-style-type: none"> A query is executed 	<ul style="list-style-type: none"> Execute a query
<ul style="list-style-type: none"> A query is edited 	<ul style="list-style-type: none"> Edit a query
<ul style="list-style-type: none"> The query results are provided in different forms. <i>Range: data Set, form, report</i> 	<ul style="list-style-type: none"> Provide the query results in different forms.
ASSESSMENT TASKS OR ACTIVITIES	
<ul style="list-style-type: none"> Class tests Projects/Practical work Assignments Group work 	

SUBJECT OUTCOME	
5.5 Retrieve information from a database by applying a filter.	
ASSESSMENT STANDARD	LEARNING OUTCOME
<ul style="list-style-type: none"> A filter is created 	<ul style="list-style-type: none"> Create a filter
<ul style="list-style-type: none"> A filter is applied to the database table to filter out specific records. 	Apply a filter to the database to filter out specific records.
<ul style="list-style-type: none"> A filter is removed. 	<ul style="list-style-type: none"> Demonstrate how to remove a filter.
ASSESSMENT TASKS OR ACTIVITIES	
<ul style="list-style-type: none"> Class tests Projects/Practical work Assignments Group work 	

SUBJECT OUTCOME	
5.6 Create a report.	
ASSESSMENT STANDARD	LEARNING OUTCOME
<ul style="list-style-type: none"> A report is designed to address the requirement of the given problem. 	<ul style="list-style-type: none"> Design a report to address the requirements of a given problem.
<ul style="list-style-type: none"> A report is created according to the design. 	<ul style="list-style-type: none"> Create a report according to the design
<ul style="list-style-type: none"> A report is modified 	<ul style="list-style-type: none"> Modify a report
<ul style="list-style-type: none"> Data is grouped within a report. 	<ul style="list-style-type: none"> Group data within a report.
<ul style="list-style-type: none"> A report is saved 	<ul style="list-style-type: none"> Save a report
<ul style="list-style-type: none"> A report is edited. 	<ul style="list-style-type: none"> Edit a report.

ASSESSMENT TASKS OR ACTIVITIES
<ul style="list-style-type: none"> • Class tests • Projects/Practical work • Assignments • Group work

SUBJECT OUTCOME	
5.7 Perform advanced print options for a database.	
ASSESSMENT STANDARD	LEARNING OUTCOME
<ul style="list-style-type: none"> • A database form is printed. 	<ul style="list-style-type: none"> • Print a database form.
<ul style="list-style-type: none"> • Results of a query are printed. 	<ul style="list-style-type: none"> • Print results of a query.
<ul style="list-style-type: none"> • A database report is previewed to ensure the presentation meets the given specification. 	<ul style="list-style-type: none"> • Preview a database report to ensure the presentation meets the given specification.
<ul style="list-style-type: none"> • A database report is printed. 	<ul style="list-style-type: none"> • Print a database report.
ASSESSMENT TASKS OR ACTIVITIES	
<ul style="list-style-type: none"> • Class tests • Projects/Practical work • Assignments • Group work 	

Topic 6: Error handling in a computer programming environment

SUBJECT OUTCOME	
6.1 Explain different errors and apply debugging techniques.	
ASSESSMENT STANDARD	LEARNING OUTCOME
<ul style="list-style-type: none"> • The difference between a logical error and a syntax errors is explained 	<ul style="list-style-type: none"> • Explain the difference between a logical error and a syntax error.
<ul style="list-style-type: none"> • Different debugging techniques are explained and applied. <i>Range: trace tables, compiler debugging tools, watches, stop breaks, control stops</i> 	<ul style="list-style-type: none"> • Explain and apply different debugging techniques.
ASSESSMENT TASKS OR ACTIVITIES	
<ul style="list-style-type: none"> • Class tests • Projects/Practical work • Assignments • Group work 	

SUBJECT OUTCOME	
6.2 Explain and apply the concept of data validation and data validation techniques.	
ASSESSMENT STANDARD	LEARNING OUTCOME
<ul style="list-style-type: none"> Different input errors are identified and explained as causes of computer input devices or human errors 	<ul style="list-style-type: none"> Identify and explain different input errors.
<ul style="list-style-type: none"> Sources of induced errors in calculations are identified and explained 	<ul style="list-style-type: none"> Identify and explain sources of induced errors in calculations.
<ul style="list-style-type: none"> Data validation techniques to limit input errors are explained 	<ul style="list-style-type: none"> Explain data validation techniques to limit input errors.
<ul style="list-style-type: none"> Data validation techniques to limit input errors are implemented <i>Range: validation rules like range checks, masks, checksums</i> 	<ul style="list-style-type: none"> Implement data validation techniques to limit input errors.
ASSESSMENT TASKS OR ACTIVITIES	
<ul style="list-style-type: none"> Class tests Projects/Practical work Assignments Group work 	

SUBJECT OUTCOME	
6.3 Demonstrate how calculation errors are induced in the computer.	
ASSESSMENT STANDARD	LEARNING OUTCOME
<ul style="list-style-type: none"> The demonstration explains overflow errors. <i>Range: overflow, underflow, conversion errors</i> 	<ul style="list-style-type: none"> Explain and demonstrate overflow errors.
<ul style="list-style-type: none"> The demonstration explains underflow errors. 	<ul style="list-style-type: none"> Explain and demonstrate underflow errors.
<ul style="list-style-type: none"> The demonstration explains conversion errors. <i>Range: storing wrong variable types, declared fields, mixing variable types in an expression.</i> 	<ul style="list-style-type: none"> Explain and demonstrate conversion errors.
<ul style="list-style-type: none"> The demonstration explains errors found in computers as a result of advances in processor word-sizes. <i>Range: size of a value stored in a variable.</i> 	<ul style="list-style-type: none"> Explain and demonstrate errors found in computers as a result of advances in processor word-sizes.
ASSESSMENT TASKS OR ACTIVITIES	
<ul style="list-style-type: none"> Class tests Projects/Practical work Assignments Group work 	

Topic 7: User interface and output design

SUBJECT OUTCOME	
7.1 Explain and implement user interface and output design concepts	
ASSESSMENT STANDARD	LEARNING OUTCOME
<ul style="list-style-type: none"> User interface design concepts and principles are explained. <i>Range: user interface design, human computer interaction, principles of user centred design</i> 	<ul style="list-style-type: none"> Explain user interface design concepts and principles.
<ul style="list-style-type: none"> The guidelines for user interface design are listed. 	<ul style="list-style-type: none"> List the guidelines for user interface design.
<ul style="list-style-type: none"> Various user interface techniques are described. <i>Range: screen elements, control</i> 	<ul style="list-style-type: none"> Describe various user interface techniques.

<ul style="list-style-type: none"> • Input design concepts, techniques and methods are discussed. 	<ul style="list-style-type: none"> • Discuss input design concepts, techniques and methods.
<ul style="list-style-type: none"> • Output design issues and various types of output are discussed. 	<ul style="list-style-type: none"> • Discuss output design issues and various types of output.
ASSESSMENT TASKS OR ACTIVITIES	
<ul style="list-style-type: none"> • Class tests • Projects/Practical work • Assignments • Group work 	

SUBJECT OUTCOME	
7.2 Explain and apply defensive programming.	
ASSESSMENT STANDARD	LEARNING OUTCOME
<ul style="list-style-type: none"> • A solution is developed implementing user interface and output design principles 	<ul style="list-style-type: none"> • Develop a solution implementing user interface and output design principles.
<ul style="list-style-type: none"> • Defensive programming is explained 	<ul style="list-style-type: none"> • Explain defensive programming.
<ul style="list-style-type: none"> • Typical human behaviour that necessitates defensive programming is anticipated 	<ul style="list-style-type: none"> • Anticipate typical human behaviour that necessitates defensive programming.
<ul style="list-style-type: none"> • Defensive programming techniques are applied 	<ul style="list-style-type: none"> • Apply defensive programming techniques
ASSESSMENT TASKS OR ACTIVITIES	
<ul style="list-style-type: none"> • Class tests • Projects/Practical work • Assignments • Group work 	

4 SPECIFICATIONS FOR EXTERNAL ASSESSMENT IN PRINCIPLES OF COMPUTER PROGRAMMING - LEVEL 3

4.1 Integrated summative assessment task (ISAT)

A compulsory component of the external assessment (ESASS) is the **integrated summative assessment task (ISAT)**. The integrated summative assessment task (ISAT) draws on the students' cumulative learning achieved throughout the year. The task requires **integrated application of competence** and is executed and recorded in compliance with assessment conditions.

Two approaches to the integrated summative assessment task (ISAT) may be as follows:

- The students are assigned a task at the beginning of the year which they will have to complete in phases during the year to obtain an assessment mark. A final assessment is made at the end of the year when the task is completed.

OR

- Students achieve the competencies during the year but the competencies are assessed cumulatively in a single assessment or examination session at the end of the year.

The integrated summative assessment task (ISAT) is set by an externally appointed examiner and is conveyed to colleges in the first quarter of the year.

The integrated assessment approach enables students to be assessed in more than one subject with the same integrated summative assessment task (ISAT).

4.2 National Examination

A national examination is conducted annually in October or November by means of a paper(s) set and moderated externally. The following distribution of cognitive application is suggested:

LEVEL 3	KNOWLEDGE AND COMPREHENSION	APPLICATION	ANALYSIS, SYNTHESIS AND EVALUATION
	40%	40%	20%