Unit Title:	**Data Structures and Algorithms**
OCR unit number	41
Level:	4
Credit value:	15
Guided learning hours:	90
Unit reference number:	R/601/3297

Candidates undertaking this unit must complete real work activities in a work environment. Simulation is only allowed in exceptional circumstances (please refer to the centre handbook for further details).

Unit purpose and aim

To provide the learner with the skills and competencies in the use of data structures in information systems, together with the algorithms associated with them.

Learning Outcomes	**Assessment Criteria**	**Knowledge, understanding and skills**
The Learner will: 1 Understand the structure of various data structures and their associated algorithms	**The Learner can:** 1.1 Define the technology used to describe the elements of data structures including arrays, linked lists, stacks, queues, trees, graphs and sets 1.2 Explain how one-dimensional and multi-dimensional arrays are structured and processed 1.3 Explain how linked lists (including singly, doubly and circular linked lists) are structured and processed 1.4 Explain how stacks and queues are structured and processed 1.5 Explain how trees and graphs are structured and processed. 1.6 Explain how sets are structured and processed	Candidates must have a detailed understanding of: The logical models of: • arrays • linked lists • stacks • trees • graphs • sets. The physical models of, and processing methods for: • arrays • linked lists • stacks • queues • trees • graphs • sets
2 Understand the operation of established algorithms	2.1 Explain the operation and performance of sorting and searching algorithms 2.2 Explain the operation of recursive algorithms and identify situations when recursion is used	Candidates must have a detailed understanding of: • sorting and searching algorithms (e.g. insertion sort, selection sort, bubble sort, shell sort, heap sort), • recursive algorithms (tail and non-tail recursion).

Learning Outcomes	Assessment Criteria	Knowledge, understanding and skills
3 Select appropriate data structures and associated algorithms for specified problems	3.1 Given a specified problem, choose a data structure and associated algorithm and justify the selection	Potential range of problems include: • heap sort • recursive interpreter • infix to postfix conversion (stacks) • binary tree transversal
4 Describe the data structures and associated algorithms in a non-executable program specification language	4.1 Specify the structure and associated algorithms of arrays, linked lists, stacks, queues, trees, graphs and sets in well-established specification languages 4.2 Specify the behaviour of sorting, searching and recursive algorithms using well-established specification languages 4.3 Demonstrate the operation of data structures and algorithms by hand execution of the associated algorithms with specified test data	Candidates must have a knowledge of a specification language (e.g. Z, VDM, Larch) and use it to define the structure and algorithms to process: • arrays • linked lists • stacks • queues • trees • graphs • sets Use such structures to develop algorithms to sort, search and process recursively. Dry run such algorithms using appropriate test data.
5 Implement data structures and algorithms in an executable programming language	5.1 Implement arrays, linked lists, stacks, queues, trees, graphs and sets in the context of well-defined problems in an executable programming language. 5.2 Implement sorting, searching and recursive algorithms arrays in the context of well-defined problems in an executable programming language. 5.3 Demonstrate the correct operation of data structure algorithms implemented in an executable programming language by devising and executing testing strategies.	Candidates must have a knowledge of an appropriate computer programming language (e.g. C++, C#, java) and use it to define the structure, algorithms and definition of executable programs to process: • linked lists • stacks • queues • trees • sets Use such structures to develop algorithms to sort, search and process recursively. Devise appropriate testing schedules.
6 Understand how strings are structured and processed	6.1 Explain the structure of strings 6.2 Identify common string operations 6.3 Demonstrate the outcome of string operations on specified strings	Candidates must have a knowledge of: • structure of strings • string operations(e.g. exact matching, approximate matching, alphabetic sorting, copying)

Assessment

Candidates undertaking this unit must complete real work activities in order to produce evidence to demonstrate they are occupationally competent. Real work is where the candidate is engaged in activities that contribute to the aims of the organisation by whom they are employed, for example in paid employment or working in a voluntary capacity.

Simulation is only allowed for aspects of units when a candidate is required to complete a work activity that does not occur on a regular basis and therefore opportunities to complete a particular work activity do not easily arise. When simulation is used, assessors must be confident that the simulation replicates the workplace to such an extent that candidates will be able to fully transfer their occupational competence to the workplace and real situations.

Internal quality assurance personnel must agree the use of simulated activities before they take place and must sample all evidence produced through simulated activities.

It is the assessor's role to satisfy themselves that evidence is available for all performance, knowledge and evidence requirements before they can decide that a candidate has finished a unit. Where performance and knowledge requirements allow evidence to be generated by other methods, for example by questioning the candidate, assessors must be satisfied that the candidate will be competent under these conditions or in these types of situations in the workplace in the future. Evidence of questions must include a written account of the question and the candidate's response. Observations and/or witness testimonies must be detailed and put the evidence into context ie the purpose of the work etc.

All of the assessment criteria in the unit must be achieved and clearly evidenced in the submitted work, which is externally assessed by OCR.

Evidence for the knowledge must be explicitly presented and not implied through other forms of evidence.

Evidence requirements

All aspects of the assessment criteria must be covered and evidence must be available that shows where and how the assessment criteria have been achieved.

Assessment Criterion 1

Candidates must:

- define the terminology used to describe logical models which meet the design specifications for the stated structures of: arrays, linked lists stacks, trees graphs and sets.
- explain how the logical model can be implemented as a physical data structure with in a computer program.

Assessment Criterion 2

Candidates must:

- define at least 3 algorithms for sorting/searching data.

Based on a recursive definition, candidates must define an appropriate recursive algorithm to process that definition.

Assessment Criterion 3

Candidates must select the data structure and associated algorithm for a non-trivial example, justifying choices made.

Assessment Criterion 4

Using an appropriate specification language, candidates must define:

- physical data structures for arrays, linked lists, stacks, queues, trees, graphs and sets.
- sorting / searching and recursive routines

Based on these specifications, produce appropriate documentation of desk testing the defined algorithms, using appropriate data.

Assessment Criterion 5

Based on the specifications of assessment criteria 4, the candidate must further develop these into executable programs, using an appropriate programming language.

Candidate to demonstrate operation of these programs, using appropriate test data.

Assessment Criterion 6

Candidates must:

- explain how strings are structured
- define standard string operations
- develop appropriate executable code to perform such operations.

Candidates are encouraged to choose activities which will allow them to cover all or a majority of the criteria at one time. It is not necessary to use different activities for each element of the criterion.

Guidance on assessment and evidence requirements

Evidence can reflect how the candidate carried out the process or it can be the product of a candidate's work or a product relating to the candidate's competence.

For example: The process that the candidate carries out could be recorded in a detailed personal statement or witness testimony. It is the assessor's responsibility to make sure that the evidence a candidate submits for assessment meets the requirements of the unit.

Questioning the candidate is normally an ongoing part of the assessment process, and is necessary to:

- test a candidate's knowledge of facts and procedures
- check if a candidate understands principles and theories *and*
- collect information on the type and purpose of the processes a candidate has gone through.
- Candidate responses must be recorded

It is difficult to give a detailed answer to how much evidence is required as it depends on the type of evidence collected and the judgement of assessors. The main principles, however, are as follows: for a candidate to be judged competent in a unit, the evidence presented must satisfy:

- all the items listed, in the section 'Learning Outcomes'
- all the areas in the section 'Assessment Criteria'

The quality and breadth of evidence provided should determine whether an assessor is confident that a candidate is competent or not. Assessors must be convinced that candidates working on their own can work independently to the required standard.

You should refer to the '*Admin Guide: Vocational Qualifications* (A850)' for *Notes on Preventing Computer-Assisted Malpractice*.

Resources

Books

Goodrich M, Tamassia R – *Data Structures and Algorithms in C++* (Wiley, 2010)
ISBN-10: 0470383275

Goodrich M, Tamassia R – *Data Structures and Algorithms in Java* (Wiley, 2005)
ISBN-10: 0471738840

Melhorn K, Sanders P – *Algorithms and Data Structures*: *The Basic Toolbox* (Springer, 2008)
ISBN-10: 3540779779

Drozdek A – Data Structures and Algorithms in C++ (Thomson, 2005)
ISBN 0-534-49182-0

Drozdek A – Data Structures and Algorithms in Java (Thomson, 2004)
ISBN 0-534-49252-5

Additional information

For further information regarding administration for this qualification, please refer to the OCR document '*Admin Guide: Vocational Qualifications' (A850)* on the OCR website www.ocr.org.uk