

Pearson Higher Nationals in Computing

Unit 19: Data Structures & Algorithm

Issue 1



Higher National Certificate/Diploma in Computing

Assignment Brief

Student Name/ID Number	
Unit Number and Title	19: Data Structures & Algorithm
Academic Year	
Unit Tutor	
Assignment Title	Data manipulation in different environment
Issue Date	
Submission Date	
IV Name & Date	

Submission Format

The submission of the following forms as a one word processing document :

1. Part 1 :
 - a. A presentation
 - b. Illustrated document
 - c. 1000 words report
2. Part 2 : Program source code with relevant screen captures of running program.
3. Part 3 :
 - a. Documentation
 - b. Program source code with relevant screen captures of running program.

You are required to make use of headings, paragraphs, subsections and illustrations as appropriate, and theoretical sections can be supported with research and (where appropriate) referenced using the Harvard referencing system.

Unit Learning Outcomes

- LO1. Examine abstract data types, concrete data structures and algorithms.
- LO2. Specify abstract data types and algorithms in a formal notation.
- LO3. Implement complex data structures and algorithms.
- LO4. Assess the effectiveness of data structures and algorithms.

Assignment Brief and Guidance

Part 01:

a. Assume you are a software engineer in a system software development company and the tech lead of your team asks you to make a 30 minutes presentation for the trainee programmers on the topic of Binary Search Trees versus other data structures including the speaker notes, when required. **(LO1,LO4)**

b. Demonstrate the following values in a BST with a simple diagram and write algorithms for inserting and traversing the tree with step by step illustration.

12,34,23,56,34,78,45,78,23,78,89,4,57,3,8,9 **(LO1, LO2)**

c. Write a 1000 words report by comparing and contrasting the elementary sorting algorithms with advanced sorting algorithms. **(LO1,LO4)**

Part 02:

An Operation theatre needs an application for storing details of patients who undergo operations daily. Nurses of the wards may insert the patient details such as admission number, name, ward no., age, type of operation (Heart, Eye, Head ..etc.), blood pressure , sugar level and so on to the application and doctors may get the patients to the operation theatre based on the order in which they have been inserted to the system. Implement the program using a data structure you prefer explaining reasons why you selected that in your application. You can use any programming language you like. **(LO1,LO3)**

Part 03:

a. Use a formal methods notation to design the following operations.

1. Create 1000 elements integer array.
2. Populate the array with random numbers within the range of 100 to 1000.
3. Sort the array using insertion sorting technique.
4. Search the array for the values 857,235,78,567,234,165,1020 using binary search algorithm. **(LO1,LO2)**

b. Using a programming language implement the above operations and if the given values are not found test your program until it finds a value. **(LO1,LO3)**

Learning Outcomes and Assessment Criteria

Pass		Merit	Distinction
LO1 Examine abstract data types, concrete data structures and algorithms			D1 Analyse the operation, using illustrations, of two network shortest path algorithms, providing an example of each.
P1 Create a design specification for data structures explaining the valid operations that can be carried out on the structures.	P2 Determine the operations of a memory stack and how it is used to implement function calls in a computer.	M1 Illustrate, with an example, a concrete data structure for a First In First out (FIFO) queue. M2 Compare the performance of two sorting algorithms.	
LO2 Specify abstract data types and algorithms in a formal notation			D2 Discuss the view that imperative ADTs are a basis for object orientation and, with justification, state whether you agree.
P3 Using an imperative definition, specify the abstract data type for a software stack.	P4 Using an imperative definition, specify the abstract data type for a software stack.	M3 Examine the advantages of encapsulation and information hiding when using an ADT.	
LO3 Implement complex data structures and algorithms			D3 Critically evaluate the complexity of an implemented ADT/algorithm.
P4 Implement a complex ADT and algorithm in an executable programming language to solve a well-defined problem.	P5 Implement error handling and report test results.	M4 Demonstrate how the implementation of an ADT/algorithm solves a well-defined problem.	

Pass	Merit	Distinction
LO4 Assess the effectiveness of data structures and algorithms		D4 Evaluate three benefits of using implementation independent data structures.
P6 Discuss how asymptotic analysis can be used to assess the effectiveness of an algorithm. P7 Determine two ways in which the efficiency of an algorithm can be measured, illustrating your answer with an example.	M5 Interpret what a trade-off is when specifying an ADT using an example to support your answer.	