**ASSIGNMENT 1 FRONT SHEET**

|  |  |  |  |
| --- | --- | --- | --- |
| **Qualification** | **BTEC Level 5 HND Diploma in Computing** | | |
| **Unit number and title** | Unit 19: Data Structures and Algorithms | | |
| **Submission date** |  | **Date Received 1st submission** |  |
| **Re-submission Date** |  | **Date Received 2nd submission** |  |
| **Student Name** | Nguyen Anh Trinh | **Student ID** | GCC210105 |
| **Class** | GCC1001 | **Assessor name** | Tran Thi Kim Khanh |
| **Student declaration**  I certify that the assignment submission is entirely my own work and I fully understand the consequences of plagiarism. I understand that making a false declaration is a form of malpractice. | | | |
|  |  | **Student’s signature** |  |

**Grading grid**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **P1** | **P2** | **P3** | **M1** | **M2** | **M3** | **D1** | **D2** |
|  |  |  |  |  |  |  |  |

|  |  |  |
| --- | --- | --- |
| **❒ Summative Feedback: ❒ Resubmission Feedback:** | | |
| **Grade:** | **Assessor Signature:** | **Date:** |
| **Internal Verifier’s Comments:** | | |
| **IV Signature:** | | |

# Assignment Brief 1 (RQF)

## Higher National Certificate/Diploma in Business

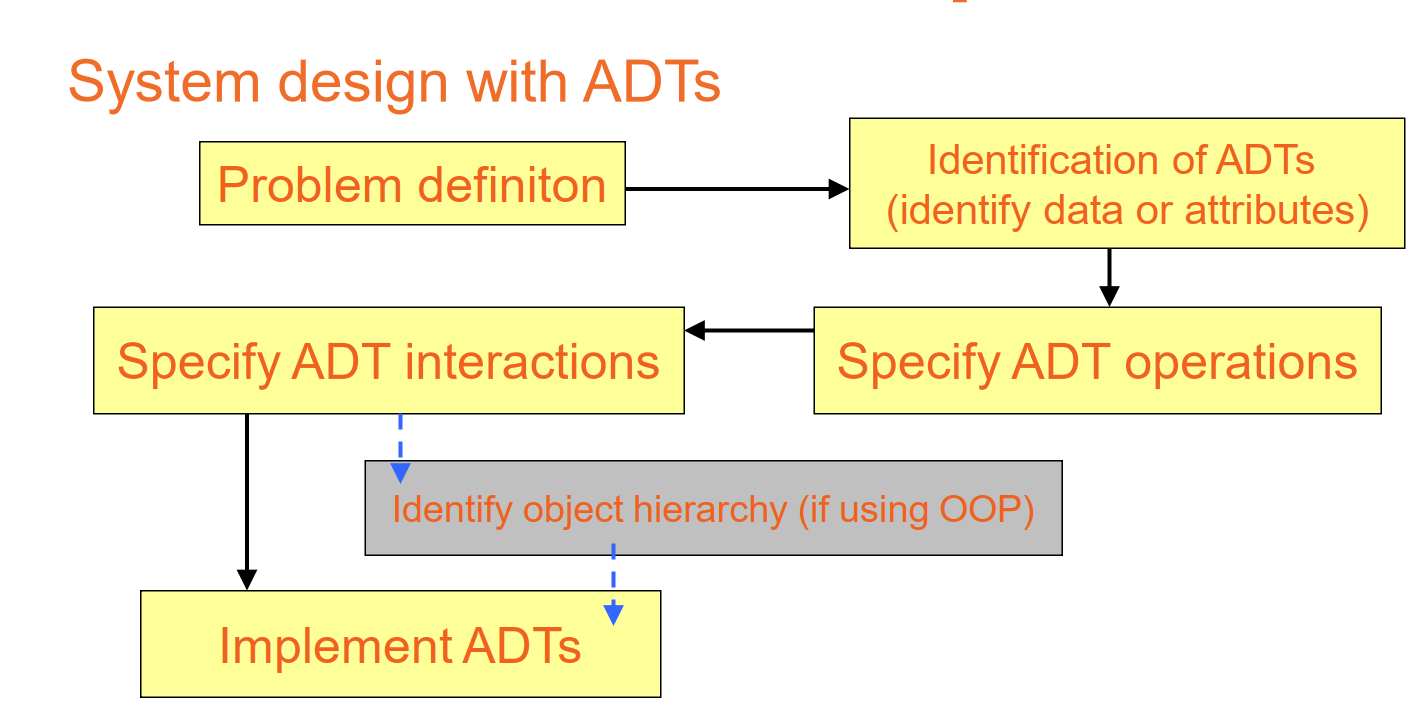
|  |  |
| --- | --- |
| **Student Name/ID Number:** |  |
| **Unit Number and Title:** | Unit 19: Data Structures and Algorithms |
| **Academic Year:** | **2021** |
| **Unit Assessor:** |  |
| **Assignment Title:** | Examine and specify ADT and DSA |
| **Issue Date:** |  |
| **Submission Date:** |  |
| **Internal Verifier Name:** |  |
| **Date:** |  |

|  |
| --- |
| **Submission Format:** |
| *Format:*   * The submission is in the form of an individual written report and a presentation. This should be written in a concise, formal business style using single spacing and font size 12. You are required to make use of headings, paragraphs and subsections as appropriate, and all work must be supported with research and referenced using the Harvard referencing system. Please also provide a bibliography using the Harvard referencing system.   *Submission*   * Students are compulsory to submit the assignment in due date and in a way requested by the Tutor. * The form of submission will be a soft copy posted on <http://cms.greenwich.edu.vn/>. * Remember to convert the word file into PDF file before the submission on CMS.   *Note:*   * The individual Assignment *must* be your own work, and not copied by or from another student. * If you use ideas, quotes or data (such as diagrams) from books, journals or other sources, you must reference your sources, using the Harvard style. * Make sure that you understand and follow the guidelines to avoid plagiarism. Failure to comply this requirement will result in a failed assignment. |
| **Unit Learning Outcomes:** |
| **LO1** Examine abstract data types, concrete data structures and algorithms  **LO2** Specify abstract data types and algorithms in a formal notation |
| **Assignment Brief and Guidance:** |
| **Assignment scenario**  You work as in-house software developer for Softnet Development Ltd, a software body-shop providing network provisioning solutions. Your company is part of a collaborative service provisioning development project and your company has won the contract to design and develop a middleware solution that will interface at the front-end to multiple computer provisioning interfaces including SOAP, HTTP, JML and CLI, and the back-end telecom provisioning network via CLI .  Your account manager has assigned you a special role that is to inform your team about designing and implementing abstract data types. You have been asked to create a presentation for all collaborating partners on how ADTs can be utilised to improve software design, development and testing. Further, you have been asked to write an introductory report for distribution to all partners on how to specify abstract data types and algorithms in a formal notation.  **Tasks**  **Part 1**  You will need to prepare a presentation on how to create a design specification for data structures, explaining the valid operations that can be carried out on the structures using the example of:   1. A stack ADT, a concrete data structure for a First In First out (FIFO) queue. 2. Two sorting algorithms. 3. Two network shortest path algorithms.   **Part 2**  You will need to provide a formal written report that includes the following:   1. Explanation on how to specify an abstract data type using the example of software stack. 2. Explanation of the advantages of encapsulation and information hiding when using an ADT. 3. Discussion of imperative ADTs with regard to object orientation. |

|  |  |  |
| --- | --- | --- |
| Learning Outcomes and Assessment Criteria (Assignment 1) | | |
| 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. | **M3** Examine the advantages of encapsulation and information hiding when using an ADT. |

# Data structures

## Abstract data type (P1)

Definition: An object's behaviour may be characterized by a set of values and a set of actions, and this behaviour is known as an abstract data type (ADT). The definition of ADT merely specifies the actions that must be used, not how they must be carried out. It is unclear what algorithms might be utilized to carry out the operations and how the data will be structured in memory. Because it presents an implementation-independent view, it is dubbed "abstract." (geeksforgeek, 2022)

The process of defining and implementing ADT includes 5 steps:

* Step 1: Clarify the nature of the target information unit by understanding it, and then apply ADT to solve it.
* Step 2: define ADT: detailed description of attributes and data types.
* Step 3: Specify ADT actions and operations: The specification of actions and operations for ADT entails performing specifications in these activities.
* Step 4: Detail ADT interactions: When using OOP and the abstract data type (ADT), there are interactions between the ACTs, inheritance of who is the father and who is the kid, and other issues that need determining the object hierarchy.
* Step 5: Adopt ADT via class (ADT is an interface) to implement ADT. how things function. For instance, an ADT may be expressed using a Java interface, which is nothing more than a set of method declarations with empty method bodies. An ADT is implemented using a real-world data structure that is represented by a Java class. Classes specify both the data that is stored and the operations that are supported by objects that are instances of the class. (Flm, 2022)

**Examples :**

Singly Linked list: A singly linked list is a linear data structure in which the elements are not stored in contiguous memory locations and each element is connected only to its next element using a pointer.



The singly linked list class has two attributes: list —the pointer to the first node in the list, and. size —an integer to keep track of the number of items in the list.

(geeksforgeeks, 2022)

* AddFirst:

