## Computing Year 3: 2023-2024

## Advanced DSA - Take Home Assignment

**INSTRUCTIONS**

* Answer all of the following questions and submit on a **single pdf document** by the due date. This submission document must have your student number in its name. The front page of this document must contain your name and your student number.
* Upload your submission to: [HERE](https://setuo365-my.sharepoint.com/:f:/g/personal/aine_byrne_setu_ie/EuauWJksSNlLt-JeH2ARzrIBE0SWZFxjMswA8V-HOHO4NQ)
* **Due Date: Sunday November 5th 12 midnight**
* You must demonstrate how you arrive at any answers you get. Correct answers without any explanations/demonstrations will get no credit.
* This assignment has an 50% credit weighting for the Advanced DSA module
* Submissions after November 5th will not be considered and a score of zero will be allocated.
* **You must do this assignment yourself**. Any group of assignment answers that I deem to be too similar for coincidence will be sent to the External Examiner for arbitration. If it is adjudged that plagiarism has occurred then **all parties involved will get a score of zero** on this assignment.
* **All work presented must be your own work.** You may consult class notes, books, online resources for this assignment. If you consult online resources, you are required to submitthe **Acknowledge, Describe, Evidence form** along with your submission.

**Question 1: (20%)**

Write the pseudocode for a non-recursive algorithm to add two numbers represented as two doubly linked lists together and store the result in another doubly linked list.  
Represent each number as a doubly linked list with each 3 digits stored as data of a node. (radix 1000)

Example- 1,065,313,750 would be represented as 4 nodes as below.

1

65

313

750

Demonstrate how your algorithm works on your chosen numbers.

**Question 2: (40%)**

1. Create a Binary Search Tree (BST) with the firstnames of ten of your friends. List the order of entry of the data to the BST.
2. Write the algorithm to output this list (from the BST) in alphabetical order.

Demonstrate how your algorithm works on your BST.

1. Write the algorithm that returns the number of names in the tree.

Demonstrate how your algorithm works on your BST.

1. Write the algorithm that output the contents of nodes which have a value greater than a given letter value.

Demonstrate how your algorithm works on your BST.

1. Write the algorithm that searches for a given FriendName in the tree. It will return whether the FriendName has been found or not. It will also return the search path used.

For example, given a tree with the following structure:

Diagram, radar chart

Description automatically generated

Searching for DAVID (from root) , would result in left, right, DAVID Found

Searching for AINE (from root) , would result in left, left, left, Not Found  
  
Demonstrate how your algorithm works on your BST.

1. List the Big O value for each of your algorithms.

**Question 3: (40%)**

Design a Project management application which handles projects and tasks within these projects. Use a linked list(of your choice) to store the project and task details.

Explain the data structure(s) and include a drawing of a node of your structure(s).

Name and describe eight methods in your app and show how they will work together by writing the pseudocode of each method.

The app should be capable of adding a task to a project, completing a task, list the next task in each project and change the order of tasks within a project (include these as 4 of your 8 methods). Methods like displayAll/viewAll, isEmpty etc.. will not be considered as suitable methods to be counted.