Computing 3: 2020-2021
DSA II- Assignment 2

Instructions:

1. Write a menu-driven application to implement a **Search Engine Optimisation (SEO)** application and it's operations as described below.

2. The solution is to be presented in a **report** and is to contain
   - Thorough description of the problem,
   - Detailed description of underlying data structures used to solve the problem (use diagrams to illustrate)
   - Pseudocode of the Algorithms used,
   - Detailed description of the two extra operations included
   - Copy of the code,
   - Description of all the methods/routines which have been used (1 line each), and
   - Test data used and sample execution screenshots of outputs produced.

3. The report, a copy of the code and an executable is to be uploaded (link tba)

4. **Due Date: Feb 22nd 2021**

The **Search Engine Optimisation (SEO)** application uses URL addresses and keywords to improve a site's search visibility as the keywords will acts as an index to a URL.
The application will store the URLs and upto 4 keywords per URL. The keywords are to be stored in a Binary Search Tree (BST) with links to the URLs which contained the keyword.
For illustration purposes, assume that a keyword will appear in maximum of 5 URL addresses.

SEO needs to be able to perform the following operations implemented using a linked structure:

put(keyword, URL): Checks if the keyword already exists in the BST.
If it isn't, it will add a keyword into the BST and a link to the URL. If the keyword does exist, the URL is just added to the URL list for that keyword.

get(keyword): will return the list of URLs which uses the given keyword.

printOrder() : will output the keywords list in alphabetical order.

deleteURL(keyword, URL) : will delete the given URL from the keyword list. Note: the keyword is NOT deleted.

Count(keyword): returns the number of URLs which reference a given keyword.

- Suggest and include **2 more operations** which would be suitable and useful for SEO application.

  All operations are to be options on a menu – provide clear instructions on User Interface as to how to use it.

Sample Marking scheme:

Report									Code		
Presentation	Description problem	Data Structs Used	Spec Extra Operations	PseudoCode	Desc Functions	Code present	Test Data	Sample Exec	Quality Code	GUI	Total
5	1	8	4	12	4	1	1	5	7	2	50