MH1403 Algorithms and Computing
Lab 3 Binary Search Tree
(Week 10, 20.03.2023 – 24.03.2023)

Submission Instructions:

1. This lab is 4% of the final grade of this course.
2. The submission deadline is 11:59PM, 27 March (Monday).
3. You need to submit the codes of Task 1 and 2 through NTULearn.

Task 1. (2 marks) Define a binary search tree class with name BST. In the binary search tree class BST, there is two methods, insertNode() and preorderTraversal(). The method insertNode() inserts a node containing the input data into the binary search tree. The method preorderTraversal() returns a list containing all the data in the binary search tree following the preorder traversal.

Create an empty binary search tree with name mybst, then insert the following data into the binary search tree: (17*i+3)%37 for i from 0 to 6. Call the function preorderTraversal() to obtain the preorder traversal of mybst. Print the preorder traversal of mybst.

Submit your code in the file task1.py

Task 2. (2 marks) Define a binary search tree class BST. In the binary search tree class, there are two methods: insertNode() and find(). The method insertNode() inserts a node containing the input data into the binary search tree. For this task, the method find() receives a string (Malay word) as input, and returns its English translation. If no node is found, find() returns None.

Note that the following code is provided in the file task2.py (provided together with this lab document):

In the file malayenglish.txt (provided together with this document), each line contains a Malay word and its English translation (the first 27 spaces are used to store a Malay word). The Malay words are not sorted. Read the file into a list malayEnglishList (each element of malayEnglishList is a list with two elements: a Malay word and its English translation).

Create an empty binary search tree called mybst. Insert every element of malayEnglishList into the binary search tree mybst. After creating the binary search tree, the program asks the user to input a Malay word, then find its English translation from the binary search tree efficiently.

Submit your code in the file task2.py . Please note that:

1. You ONLY need to define the binary search tree class BST in this task.

2. The method insertNode() in this task may be slightly different from that in Task 1.

3. The driver code provided the file task2.py is for the methods insertNode() and find() being implemented using the iterative approach. If you implemented the methods insertNode() or find() using recursive approach, you need to modify the driver code slightly when you call the methods.

4. When you are writing your code, you need to put task2.py and malayenglish.txt into the same directory so that the file malayenglish.txt can be read in task2.py .

5. If you are using Linux or macOS, you need to download malayenglish.txt to your computer, then open it and save it so that the file can be read successfully in task2.py . (The text file format is lightly different on these three operating systems: Windows, Linux and macOS.)

Some sample inputs/outputs are given below:

```
Enter a Malay word, or enter exit: jalan
jalan: a road, a way, a method; to walk

Enter a Malay word, or enter exit: bahar
bahar: the sea, the ocean, a big lake, a large river

Enter a Malay word, or enter exit: abad
```

```
abad: century

Enter a Malay word, or enter exit: pasir
pasir: sand

Enter a Malay word, or enter exit: nanyang
Cannot find the Malay word nanyang

Enter a Malay word, or enter exit: exit
Program exits.
```