Generic Elective – Computer Science  
**Data Structures Using Python   
(Semester-IV)**

**PRACTICAL FILE (2025)**

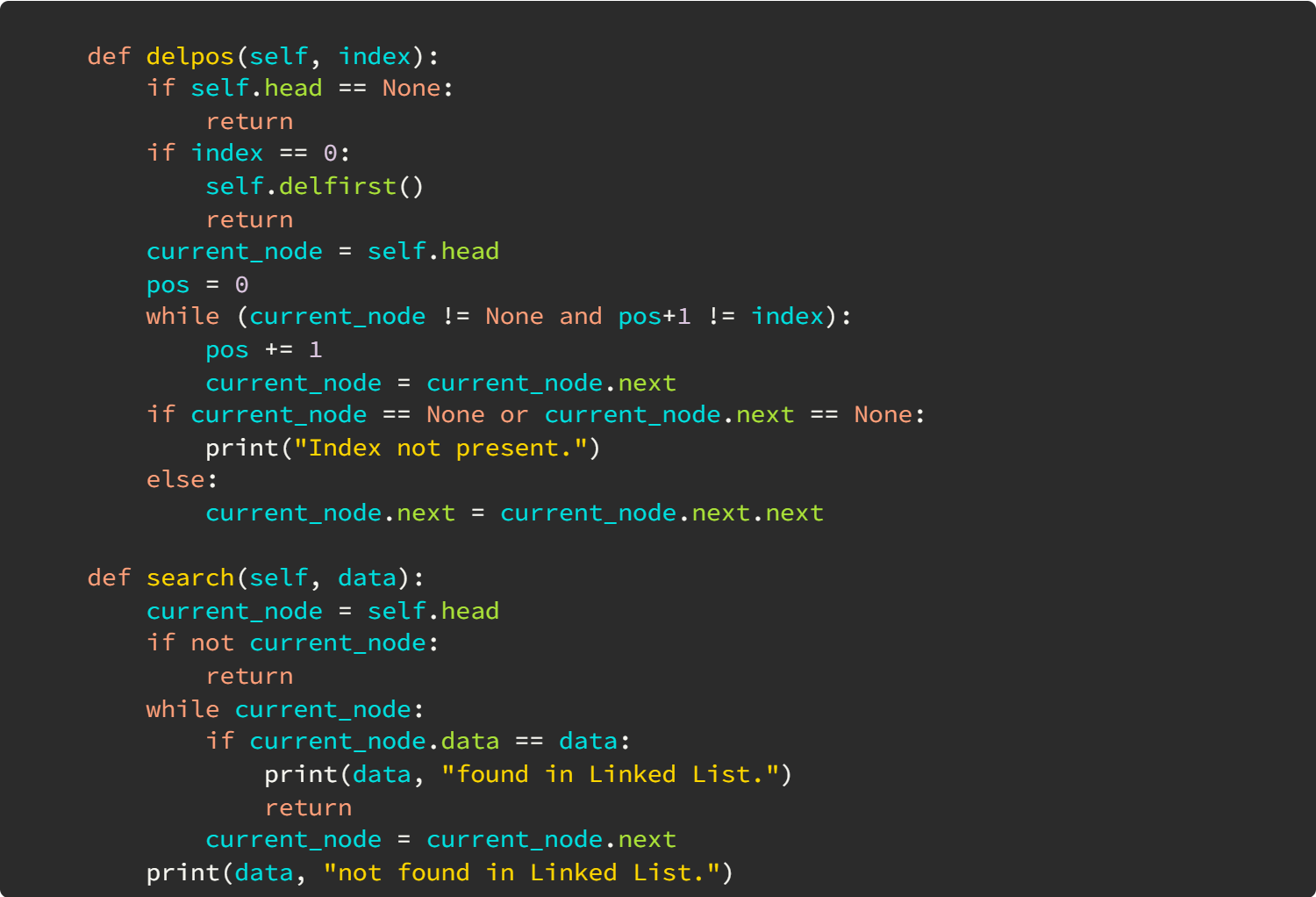
Lucky Lao  
B.Sc. (H) Mathematics  
23BMAT019

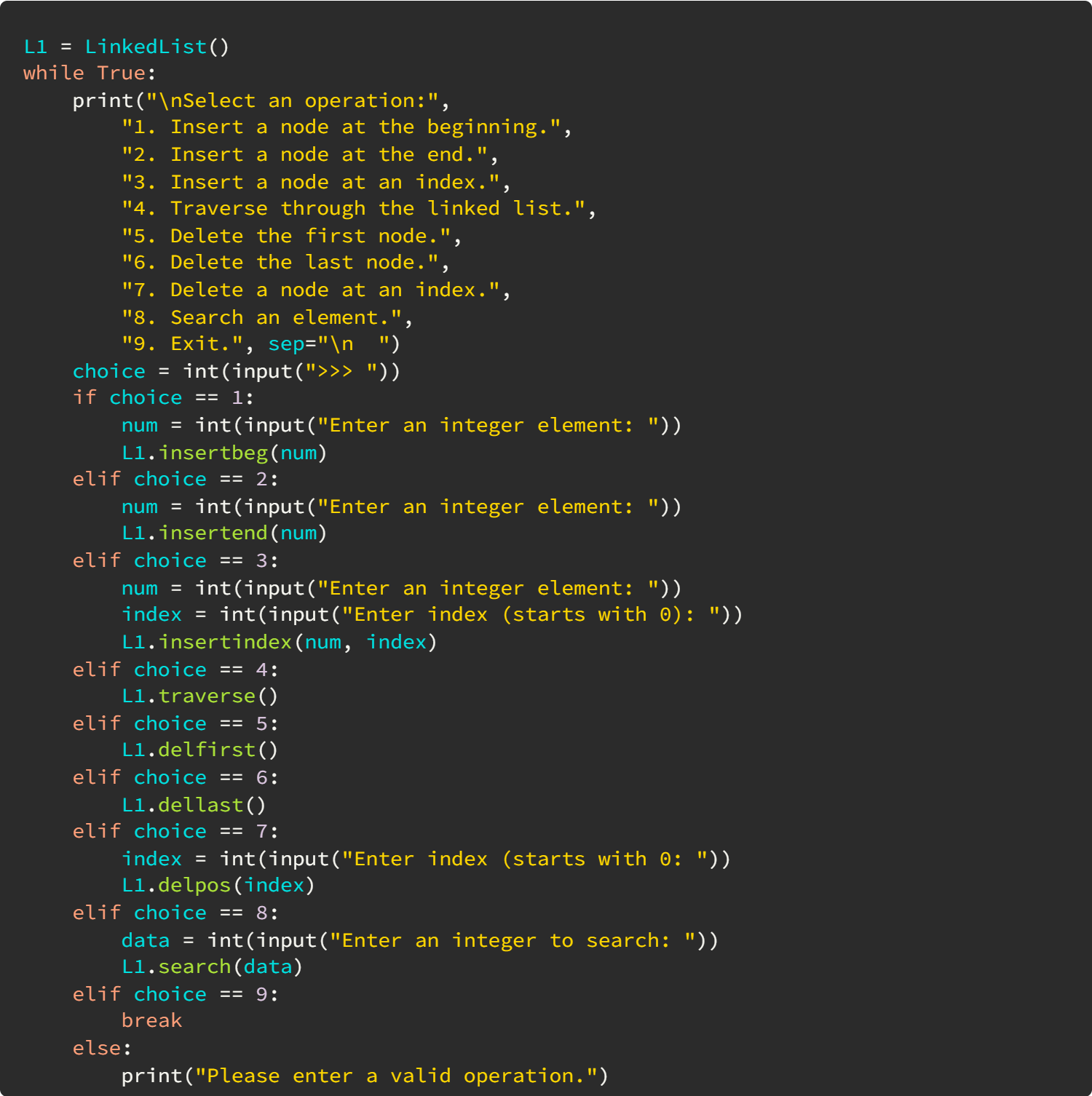
1. Write a program to implement singly linked list as an ADT that supports the following operations:
   1. Insert an element *x* at the beginning of the singly linked list.
   2. Insert an element *x* at *i*th position in the singly linked list.
   3. Remove an element from the beginning of the doubly linked list.
   4. Remove an element from *i*th position in the singly linked list.
   5. Search for an element *x* in the singly linked list and return its pointer.

***Code:***









***Output:***

1. Write a program to implement doubly linked list as an ADT that supports the following operations:
   1. Insert an element *x* at the beginning of the doubly linked list.
   2. Insert an element *x* at the end of the doubly linked list.
   3. Remove an element from the beginning of the doubly linked list.
   4. Remove an element from the end of the doubly linked list.

***Code:***

***Output:***

1. Write a program to implement circular linked list as an ADT which supports the following operations:
   1. Insert an element *x* in the list.
   2. Remove an element from the list.
   3. Search for an element *x* in the list and return its pointer.

***Code:***

***Output:***

1. Implement Stack as an ADT and use it to evaluate a prefix/postfix expression.

***Code:***

****

***Output:***

1. Implement Queue as an ADT.

***Code:***

****

***Output:***

1. Write a program to implement Binary Search Tree as an ADT which supports the following operations:
   1. Insert an element *x*.
   2. Delete an element *x*.
   3. Search for an element *x* in the BST.
   4. Display the elements of the BST in preorder, inorder, and postorder traversal.

***Code:***

***Output:***

1. Write a program to implement insert and search operation in AVL trees.

***Code:***

***Output:***