UNIVERSITY OF CALOOCAN CITY
COMPUTER ENGINEERING DEPARTMENT

Data Structure and Algorithm

Laboratory Activity No. 6

Singly Linked Lists

Submitted by:

Ruperto, April Anne A.

Instructor:

Engr. Maria Rizette H. Sayo

August 23, 2025

I. Objectives

Introduction

A linked list is an organization of a list where each item in the list is in a separate node. Linked lists look like the links in a chain. Each link is attached to the next link by a reference that points to the next link in the chain. When working with a linked list, each link in the chain is called a Node. Each node consists of two pieces of information, an item, which is the data associated with the node, and a link to the next node in the linked list, often called next.

This laboratory activity aims to implement the principles and techniques in:
- Writing algorithms using Linked list
- Writing a python program that will perform the common operations in a singly linked list

II. Methods

- Write a Python program to create a singly linked list of prime numbers less than 20. By iterating through the list, display all the prime numbers, the head, and the tail of the list. (using Google Colab)
- Save your source codes to GitHub

III. Results

Figure 1: Screenshot of program

Please follow this link:

       In this program, the Node class represents each element in the list. We start with constructor "__init__(self,data):" to initialize the data part of the node. We store the data in "self.data = data" then setting the "self.next" to None to set it by default. The class SinglyLinkedList manages the entire linked list from adding elements, printing the list, and managing the head and tail pointers. The programs iterate over numbers 2 and 19, if the number is a prime number, it will be added to the end of the linked list.

IV. Conclusion

Linked list is a linear data structure, each element is called a node. The list starts with a node called head and the last node is called tail. In Singly Linked List, you can only transversed in one direction, from head to tail. Unlike arrays, we can't directly access each element by index, and the size can grow or shrink dynamically.

References

[1] Co Arthur O.. "University of Caloocan City Computer Engineering Department Honor Code," UCC-CpE Departmental Policies, 2020.