



UNIVERSITY OF CALOOCAN CITY
COMPUTER ENGINEERING DEPARTMENT



Data Structure and Algorithm

Laboratory Activity No. 6

Singly Linked Lists

Submitted by:
Villacin, Justine R.

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

Present the visualized procedures done. Also present the results with corresponding data visualizations such as graphs, charts, tables, or image . Please provide insights, commentaries, or explanations regarding the data. If an explanation requires the support of literature such as academic journals, books, magazines, reports, or web articles please cite and reference them using the IEEE format.

Please take note of the styles on the style ribbon as these would serve as the style format of this laboratory report. The body style is Times New Roman size 12, line spacing: 1.5. Body text should be in Justified alignment, while captions should be center-aligned. Images should be readable and include captions. Please refer to the sample below:

```
0s  class Node:
    def __init__(self, data = None):
        self.data = data
        self.next = None

    def LinkedList(head):
        currentNode = head
        while currentNode:
            print(currentNode.data, end= " → ")
            currentNode = currentNode.next
        print("null")

node1 = Node(2)
node2 = Node(3)
node3 = Node(5)
node4 = Node(7)
node5 = Node(11)
node6 = Node(13)
node7 = Node(17)
node8 = Node(19)

node1.next = node2
node2.next = node3
node3.next = node4
node4.next = node5
node5.next = node6
node6.next = node7
node7.next = node8

print("Prime Numbers SinglyLinkedList: ")
LinkedList(node1)

print(f"Head: {node1.data}")
print(f"Tail: {node8.data}")

⇒ Prime Numbers SinglyLinkedList:
2 → 3 → 5 → 7 → 11 → 13 → 17 → 19 → null
Head: 2
Tail: 19
```

Figure 1 Screenshot of program

If an image is taken from another literature or intellectual property, please cite them accordingly in the caption. Always keep in mind the Honor Code [1] of our course to prevent failure due to academic dishonesty.

IV. Conclusion

This project successfully created a simple linked list program that stores and displays prime numbers below 20. Through this activity, I learned how linked lists work by connecting nodes together like a chain, where each node holds a number and points to the next one. The program effectively showed how to find the first number (head), the last number (tail), and all the numbers in between by following these connections, demonstrating the basic but powerful concept of linked list operations in programming.

References

- [1] Co Arthur O.. “University of Caloocan City Computer Engineering Department Honor Code,” UCC-CpE Departmental Policies, 2020.
- [2] W3Schools. (n.d.). *Python linked lists*. Retrieved November 27, 2023, from https://www.w3schools.com/python/python_dsa_linkedlists.asp