PROGRESS REPORT #2	
Course Code: CPE 201L	**Program:** BSCpE
Course Title: Data Structure and Algorithm	**Date Performed:** September 13, 2025
Section: 2A	**Date Submitted:** September 13, 2025
Group Members: **Leader:** Villanueva, Bryan O. **Members:** Asugas, Kenneth R. Regondola, Jezreel P. Ruperto, April Anne A. Villacin, Justine R.	**Instructor:** Engr. Maria Rizette H. Sayo

1.Objectives

This project aims to implement the following:

- Design a user friendly and visually appealing interface for the diary application.
- Integrate a doubly linked list as the underlying data structure.
- Implement core diary functions such as adding, viewing, editing, and deleting entries.

2. Discussion

During this second progress, we plan to develop a diary application in Python where users can organize their personal notes in a straightforward and organized way. The program was implemented with the intention of making the interface easy to use and visually pleasing so that users can easily interact with it like the note taking application on mobile phones. We implemented at the center of the application a Doubly Linked List (DLL) as the fundamental data structure. A diary entry is schematized as a node that holds the entry content and both previous and next entry references. This allows efficiently to navigate bidirectional navigation, users can navigate forward and backward in their diary with ease. We also added the basic diary operations new entry insertion, displaying current ones, editing errors or updates, and deleting entries when no longer necessary. With the DLL, these operations proved more efficient than with arrays since insertion and deletion do not involve shifting every element, only the pointers of the neighboring nodes are modified. This project demonstrates how data structures can be used to create useful applications that blend usability with effective management of data.

- **PC** – Personal computer use to run programs, save files, and do tasks like typing, browsing, or coding.
- **Google Colab** – An online compiler where you can write and run python code.
- **Canva** – For designing of the application.

4. Procedure

1. Choose theme for the application – As a group we decided to use minimalist style.
2. Continuation of the program – Adding functions such as viewing, editing, adding, and deleting.
3. Start of designing the application – Creating a visually appealing design for the user.
4. Naming the application - Brainstorming possible names for the application.

5. Output

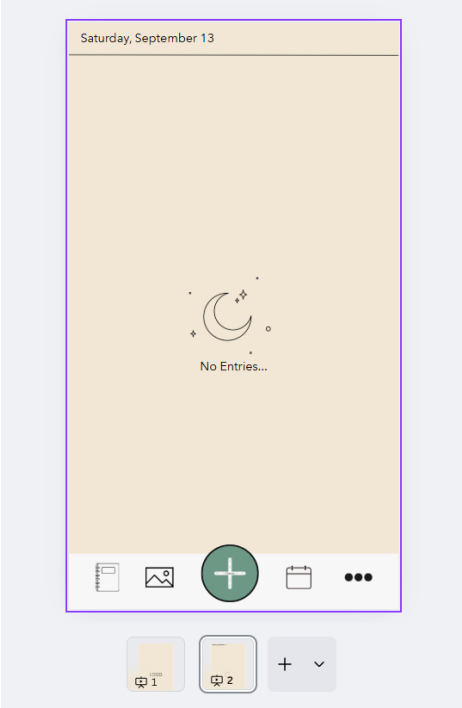

Figure 1 Design

```
def delete_by_position(self, position):
    current_node = self.head
    while current_node:
        if current_node.position == position:
            if current_node.prev is None:
                self.head = current_node.next
                if self.head:
                    self.head.prev = None
            else:
                current_node.prev.next = current_node.next
                if current_node.next:
                    current_node.next.prev = current_node.prev
            print(f"\nEntry at position {position} DELETED\n")
            self.update_positions()
            return
        current_node = current_node.next
    print("\nTHE GIVEN ENTRY IS NOT IN THE POSITION\n")

def show_all_entries(linked_list):
    print("\n--------ALL ENTRIES--------\n")
    linked_list.display()

def update_positions(self):
    current = self.head
    pos = 1
    while current:
        current.position = pos
        pos += 1
        current = current.next
```

Figure 2 Program

```python
def input_validation():
    while True:
        try:
            choice = int(input("1. Make an Entry\n"
                               "2. Delete an Entry\n"
                               "3. Show all entry\n"
                               "4. Exit\n"
                               "Enter your choice: "))
            if 1 <= choice <= 4:
                return choice
            else:
                print("\nInvalid choice! Please enter a number between 1 and 4.\n")
        except ValueError:
            print("\nInvalid input! Please enter a valid number (1-4).\n")

def make_entry(linked_list):
    entry = input("Enter your journal entry: ")
    linked_list.append(entry)
    print("\n--------ENTRY ADDED--------\n")


def delete_entry(linked_list):
    try:
        position = int(input("Enter the position of the entry to delete: "))
        linked_list.delete_by_position(position)
    except ValueError:
        print("Invalid input. Please enter a number.\n")

if __name__ == "__main__":
    dll = DoublyLinkedList()
    while True:
        choice = input_validation()

        if choice == 1:
            make_entry(dll)
        elif choice == 2:
            delete_entry(dll)
        elif choice == 3:
            show_all_entries(dll)
        elif choice == 4:
            print("Exiting the program.")
            break
```

Figure 3 Program

```
1. Make an Entry
2. Delete an Entry
3. Show all entry
4. Exit
Enter your choice: 1
Enter your journal entry: hello world

--------ENTRY ADDED--------

1. Make an Entry
2. Delete an Entry
3. Show all entry
4. Exit
Enter your choice: 3
--------ALL ENTRIES--------

Position: 1 | Date: 2025-09-13 | Entry: I ate breakfast today


Position: 2 | Date: 2025-09-13 | Entry: hello world

1. Make an Entry
2. Delete an Entry
3. Show all entry
4. Exit
Enter your choice: 2
Enter the position of the entry to delete: 1

Entry at position 1 DELETED

1. Make an Entry
2. Delete an Entry
3. Show all entry
4. Exit
Enter your choice: 3
--------ALL ENTRIES--------

Position: 1 | Date: 2025-09-13 | Entry: hello world

1. Make an Entry
2. Delete an Entry
3. Show all entry
4. Exit
Enter your choice: 4
Exiting the program.
```

Figure 4 Output of program

6. Conclusion

As a group, we were able to apply what we learned in Data Structures by creating a diary application using a Doubly Linked List. Through this project, we practiced teamwork and problem solving while building a program that is both useful and easy to use. The DLL helped us handle entries more efficiently compared to arrays, especially when adding or deleting notes. We also started working on the design to make the application simple and user-friendly. Overall, this progress showed us how data structures can be applied in real projects and gave us more confidence to continue improving our application.

Lab Activity Rubric

Criteria	Ratings						Pts
SO 7 PI 1 **Student Outcome 7.1** Acquire and apply new knowledge from outside sources. threshold: 4.8 pts	6 pts Excellent | Educational interests and pursuits exist and flourish outside classroom requirements,knowledge and/or experiences are pursued independently and applies knowledge learned into practice	5 pts Good | Educational interests and pursuits exist and flourish outside classroom requirements,knowledge and/or experiences are pursued independently	4 pts Satisfactory | Look beyond classroom requirements, showing interest in pursuing knowledge independently	3 pts Unsatisfactory | Begins to look beyond classroom requirements, showing interest in pursuing knowledge independently	2 pts Poor | Relies on classroom instruction only	1 pts Very Poor | No initiative or interest in acquiring new knowledge	6 pts
SO 7 PI 2 **Student Outcome 7.2** Learn independently threshold: 4.8 pts	6 pts Excellent | Completes an assigned task independently and practices continuous improvement	5 pts Good | Completes an assigned task without supervision or guidance	4 pts Satisfactory | Requires minimal guidance to complete an assigned task	3 pts Unsatisfactory | Requires detailed or step-by-step instructions to complete a task	2 pts Poor | Shows little interest to complete a task independently	1 pts Very Poor | No interest to complete a task independently	6 pts
SO 7 PI 3 **Student Outcome 7.3** Critical thinking in the broadest context of technological change threshold: 4.8 pts	6 pts Excellent | Synthesizes and integrates information from a variety of sources; formulates a clear and precise perspective; draws appropriate conclusions	5 pts Good | Evaluate information from a variety of sources; formulates a clear and precise perspective.	4 pts Satisfactory | Analyze information from a variety of sources; formulates a clear and precise perspective.	3 pts Unsatisfactory | Apply the gathered information to formulate the problem	2 pts Poor | Gather and summarized the information from a variety of sources but failed to formulate the problem	1 pts Very Poor | Gather information from a variety of sources	6 pts
SO 7 PI 4 **Student Outcome 7.4** Creativity and adaptability to new and emerging technologies threshold: 4.8 pts	6 pts Excellent | Ideas are combined in original and creative ways in line with the new and emerging technology trends to solve a problem or address an issue.	5 pts Good | Ideas are creative and adapt the new knowledge to solve a problem or address an issue	4 pts Satisfactory | Ideas are creative in solving a problem, or address an issue	3 pts Unsatisfactory | Shows some creative ways to solve the problem	2 pts Poor | Shows initiative and attempt to develop creative ideas to solve the problem	1 pts Very Poor | Ideas are copied or restated from the sources consulted	6 pts
							Total Points: 24