LONG QUIZ ("DATA SRUCTURE SKILL TEST")	
Course Code: CPE 201L	**Program: Computer Engineering**
Course Title: Data Structure And Algorithm	**Date Performed: August 30, 2025**
Section: 2-B	**Date Submitted: August 30, 2025**
Name: Kyle Andrey D. Enverzo	**Instructor: ENGR. Maria Rizette Sayo**

1. Objectives

- To understand the implementation and usage of the **Stack** data structure, specifically focusing on how the **push** and **pop** operations work in Python.
- To demonstrate how to manipulate a string (in this case, a full name) by inserting an underscore and then traversing it using a stack, which showcases the **Last-In-First-Out (LIFO)** principle.
- To learn how to reverse the order of characters in a string by using the stack's **pop** operation, showing how the stack retrieves elements in reverse order.
- To apply Python programming concepts, including **object-oriented programming (OOP)** with the creation of a custom Stack class, and gain hands-on experience with **list manipulation** in Python.
- To enhance problem-solving skills by using a data structure to manipulate and process a real-world string, reinforcing the importance of data structures in programming.

2. Discussion

In this activity, we utilized a **Stack** data structure to manipulate a string (the full name) and demonstrate the **Last-In-First-Out (LIFO)** principle. A **stack** allows us to add elements using the push operation and remove them using the pop operation. The key concept is that the most recently added element is the first one to be removed. This principle was effectively showcased in the program.

The **program starts by defining a Stack class** with methods to handle basic stack operations like **push**, **pop**, and checking if the stack is empty. The **push method** adds a character to the stack, while the **pop method** removes the most recently added character. By using these methods, we can add characters of a full name into the stack, and later pop them out in reverse order.

In this specific implementation, **the program pushes each character of the full name ("Juan Dela Cruz") into the stack**. After all characters are pushed, an **underscore (_) is added to the stack**, demonstrating how additional elements can be inserted into the stack after all original elements. This underscore is the last element added and will be the first to be removed when we start popping the stack.

The **traverse method** was used to display the current contents of the stack without removing any elements, showing the order in which characters were pushed, with the underscore as the last item in the stack. **Finally, the program uses the pop operation to remove each character in reverse order**, starting with the underscore and moving through the other characters, demonstrating the LIFO behavior of the stack.

This program highlights the stack's primary feature of maintaining the **order of insertion** with the reverse order during **retrieval** (popping). By inserting the underscore last, we were able to show how the stack handles new elements being added at the top, with the most recent items always being popped first.

3. Materials and Equipment

- **Python Programming Language**: We used Python to implement the stack data structure and perform the necessary operations.
- **Google Colab**: An online platform was used to write, test, and run the Python program. Google Colab provides an interactive environment that supports Python and allows for easy sharing and collaboration.
- **Computer**: A set of computers was used to run the code. These computers were equipped with internet access to use Google Colab for the program execution.
- **Full Name**: The full name ("KYLE ANDREY ENVERZO") was used as input for the program, with underscores added between sections of the name.

4. Procedure

- Defined a **Stack** class with the necessary methods (push, pop, is_empty, traverse).
- Pushed each character of the full name ("KYLE ANDREY ENVERZO") into the stack one by one.
- After pushing all characters, added an underscore as the last item pushed into the stack.
- Displayed the content of the stack using the **traverse** method.
- Used the **pop** method to remove characters from the stack and print them in reverse order.

5. Output

_ O Z R E V N E Y E R D N A E L Y K

6. Conclusion

In this activity, I successfully implemented the Stack data structure in Python and demonstrated its Last-In-First-Out (LIFO) behavior. By pushing each character of a full name into the stack and inserting an underscore at the end, I was able to showcase how the stack handles the insertion and retrieval of data. The pop operation allowed the characters to be retrieved in reverse order, confirming the core functionality of the stack.

Through this exercise, I gained a deeper understanding of the push and pop operations, and how they can be used to manipulate strings in a real-world context. I also learned how to create a custom Stack class in Python, enhancing my object-oriented programming skills. This experience reinforced the importance of data structures like the stack in solving problems where data retrieval needs to follow a specific order, such as reversing a string or processing tasks in a sequential manner.

Overall, the activity not only improved my knowledge of stack operations but also deepened my understanding of how data structures can be applied in solving practical programming challenges. The program successfully achieved its goals and provided a clear demonstration of stack operations through a simple yet effective real-world example.

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	