|  |  |
| --- | --- |
| **Skill Test. <n> <title>** | |
| **Course Code:** CPE 201L | **Program:** BSCPE |
| **Course Title:** Data Structure and Algorithm | **Date Performed:** 08/30/2025 |
| **Section:** 2A | **Date Submitted:** 08/30/2025 |
| **Name:** Nerio, Hannah Grace A. | **Instructor:** Engr. Maria Rizette H. Sayo |
| 1. **Objectives** | |
| * To understand how a stack data structure works using Python * To apply stack operations such as push, pop, top and is\_empty in a simple program * To practice storing and traversing characters of a string using a stack * To demonstrate how the LIFO principle (Last in, First out) affects the order of traversal | |
| **2. Discussion** | |
| In this activity, I created a program that stores my full name inside a stack. A stack is a data structure that follows the LIFO (Last in, First out) rule, meaning the last element inserted is the first one to be removed. I used the operations push (To add characters), pop (to remove characters), top (to see the element at the top), and is\_empty (to check if the stack is empty). After pushing all the characters into the stack, I traversed the stack by continuously popping each character until it became empty. During traversal, the characters of my name appeared in reverse order because the last character pushed was the first to come out. This part of the activity clearly shows how stack traversal works and how the LIFO principle changes the order of data when removing elements. | |
| **3. Materials and Equipment** | |
| * Computer * Google Colab (used for coding and running the program) * Python (Programming language used to implement the stack) | |
| **4. Procedure** | |
| 1. Open Google Colab on the computer.  2. Create a Stack class with methods: push, pop, top, and is\_empty.  3. Store the full name "Hannah Grace A. Nerio" in a string variable.  4. Loop through the string and push each character into the stack.  5. Display the original stack before popping.  6. Traverse the stack by using a while loop that checks if the stack is not empty.  7. During traversal, pop each character one by one and print it vertically.  8. Observe that the characters appear in reverse order due to the LIFO principle. | |
| **5. Output** | |
|  | |
| **7. Conclusion**  In this activity, I learned how a stack works and how it follows the rule of Last In, First Out (LIFO). I was able to make a Python program that pushed each character of my full name into the stack. After that, I traversed the stack by popping the characters one by one, and I saw that my name was printed in reverse order. This showed me clearly how the stack structure behaves when storing and removing data. I also practiced using stack operations such as push, pop, top, and is\_empty, and I understood their purpose more. Push is for adding, pop is for removing, top is for checking the last element and is\_empty is for making sure the stack is not empty before removing. Doing this activity helped me see how these functions actually work in a program. Overall, the activity improved my understanding of stacks and traversal. It also helped me become more confident in writing code using Python. I realized that even a simple example like using my name can explain how important stacks are in programming and how they can be applied in many situations. | |
|  | |