UNIVERSITY OF CALOOCAN CITY
COMPUTER ENGINEERING DEPARTMENT

Data Structure and Algorithm

Laboratory Activity No. 8

Stacks

Submitted by:
Sorellano, John Kenneth T.

Instructor:
Engr. Maria Rizette H. Sayo

October 4, 2025

I. Objectives

Introduction

A stack is a collection of objects that are inserted and removed according to the last-in, first-out (LIFO) principle.

A user may insert objects into a stack at any time, but may only access or remove the most recently inserted object that remains (at the so-called "top" of the stack)

This laboratory activity aims to implement the principles and techniques in:

- Writing Python program using Stack
- Writing a Python program that will implement Stack operations

II. Methods

Instruction: Type the python codes below in your Colab. After running your codes, answer the questions below.

```
# Stack implementation in python


# Creating a stack
def create_stack():
    stack = []
    return stack


# Creating an empty stack
def is_empty(stack):
    return len(stack) == 0

# Adding items into the stack
def push(stack, item):
    stack.append(item)
    print("Pushed Element: " + item)

# Removing an element from the stack
def pop(stack):
    if (is_empty(stack)):
        return "The stack is empty"
    return stack.pop()

stack = create_stack()
push(stack, str(1))
push(stack, str(2))
push(stack, str(3))
push(stack, str(4))
push(stack, str(5))

print("The elements in the stack are:"+ str(stack))
```

Answer the following questions:

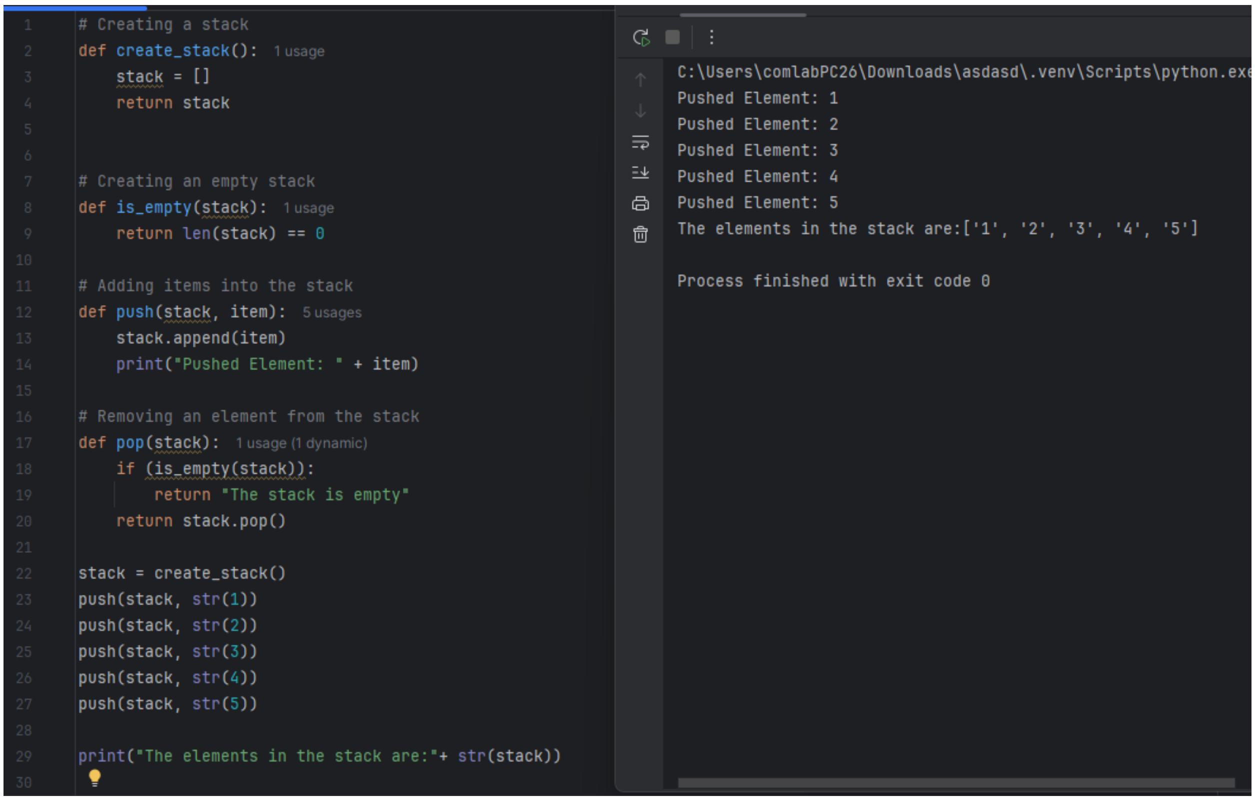
1 Upon typing the codes, what is the name of the abstract data type? How is it implemented?

2 What is the output of the codes?

3 If you want to type additional codes, what will be the statement to pop 3 elements from the top of the stack?

4 If you will revise the codes, what will be the statement to determine the length of the stack? (Note: You may add additional methods to count the no. of elements in the stack)

III. Results



```
# Creating a stack
def create_stack():
    stack = []
    return stack


# Creating an empty stack
def is_empty(stack):
    return len(stack) == 0

# Adding items into the stack
def push(stack, item):
    stack.append(item)
    print("Pushed Element: " + item)

# Removing an element from the stack
def pop(stack):
    if (is_empty(stack)):
        return "The stack is empty"
    return stack.pop()

stack = create_stack()
push(stack, str(1))
push(stack, str(2))
push(stack, str(3))
push(stack, str(4))
push(stack, str(5))

print("The elements in the stack are:"+ str(stack))
```

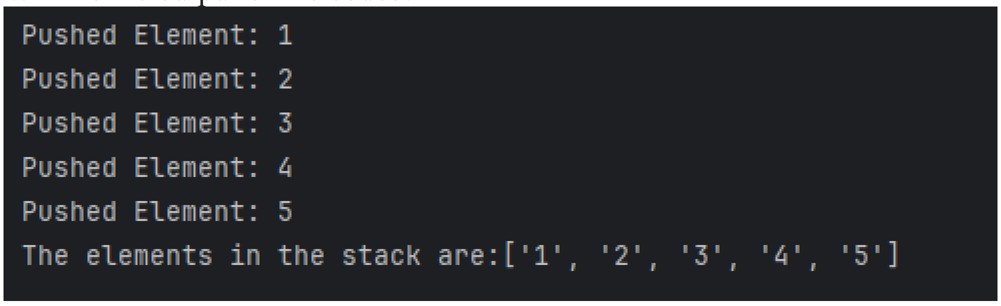
```
C:\Users\comlabPC26\Downloads\asdasd\.venv\Scripts\python.exe
Pushed Element: 1
Pushed Element: 2
Pushed Element: 3
Pushed Element: 4
Pushed Element: 5
The elements in the stack are:['1', '2', '3', '4', '5']

Process finished with exit code 0
```

1 Upon typing the codes, what is the name of the abstract data type? How is it implemented?

- The name of the abstract data type is Stack. It is implemented using the python list []. Also, the append () function used to push the element into the stack and the pop () function alternates it, it removes the element.

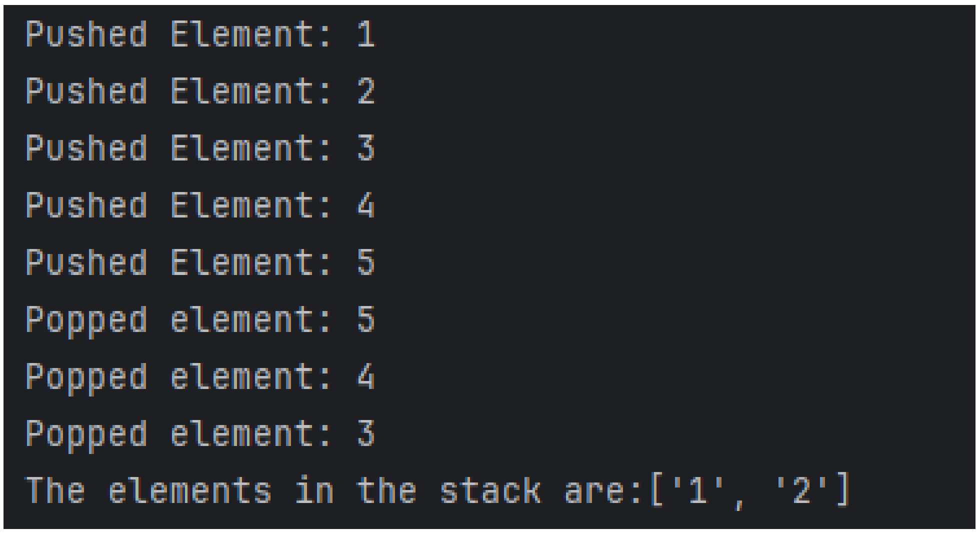
2 What is the output of the codes?



```
Pushed Element: 1
Pushed Element: 2
Pushed Element: 3
Pushed Element: 4
Pushed Element: 5
The elements in the stack are:['1', '2', '3', '4', '5']
```

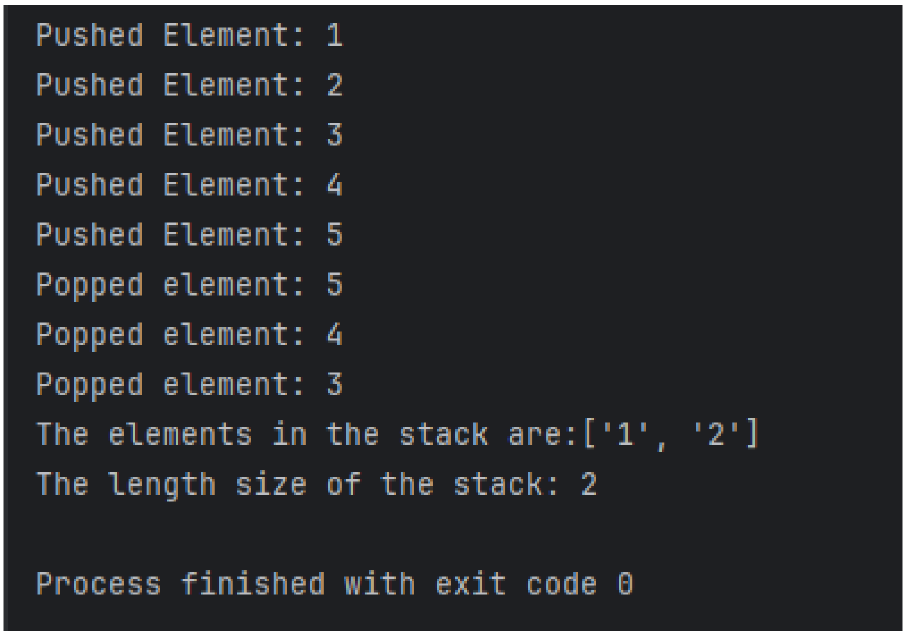
3 If you want to type additional codes, what will be the statement to pop 3 elements from the top of the stack

- If you want to pop the 3 elements from the top of the stack just add a pop function to it.



```
Pushed Element: 1
Pushed Element: 2
Pushed Element: 3
Pushed Element: 4
Pushed Element: 5
Popped element: 5
Popped element: 4
Popped element: 3
The elements in the stack are:['1', '2']
```

4 If you will revise the codes, what will be the statement to determine the length of the stack? (Note: You may add additional methods to count the no. of elements in the stack)



```
Pushed Element: 1
Pushed Element: 2
Pushed Element: 3
Pushed Element: 4
Pushed Element: 5
Popped element: 5
Popped element: 4
Popped element: 3
The elements in the stack are:['1', '2']
The length size of the stack: 2

Process finished with exit code 0
```

IV. Conclusion

This Lab report shows the abstract data type of stack using a (LIFO) function or “Last In First Out). This can be implemented by using list with append () and pop (), and many other methods like len () to check the size of the elements.

References

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