

Nama : Muhammad Roynaldi

NIM : A11.2021.13471

Kelas : A11.43UG2

## TUGAS 1

### STRUKTUR DATA

a. Link akun github

<https://github.com/muhammadroynaldi/struktur-data>

b. Source code stack dengan Bahasa pemrograman python

```
# Stack implementation in python

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

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

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

# Removing an element from the stack
def pop(stack):
    if (check_empty(stack)):
        return "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))
print("popped item: " + pop(stack))
print("stack after popping an element: " + str(stack))

```

### c. Screenshoot program stack python

The screenshot shows a Visual Studio Code editor with a file named `stack_python.py` open. The code implements a stack using a list. The terminal window on the right shows the output of running the program, which pushes four items (1, 2, 3, 4) and then pops one item (4), leaving the stack with [1, 2, 3].

```

stack_python.py
1 # Stack implementation in python
2
3
4 # Creating a stack
5 def create_stack():
6     stack = []
7     return stack
8
9
10 # Creating an empty stack
11 def check_empty(stack):
12     return len(stack) == 0
13
14
15 # Adding items into the stack
16 def push(stack, item):
17     stack.append(item)
18     print("pushed item: " + item)
19
20
21 # Removing an element from the stack
22 def pop(stack):
23     if (check_empty(stack)):
24         return "stack is empty"
25
26     return stack.pop()
27
28
29 stack = create_stack()
30 push(stack, str(1))
31 push(stack, str(2))
32 push(stack, str(3))
33 push(stack, str(4))
34 print("popped item: " + pop(stack))
35 print("stack after popping an element: " + str(stack))

```

Terminal Output:

```

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/powershell

PS C:\Users\lenovo> python -u "c:\Users\lenovo\Documents\GitHub\struktur-data\Program Stack Python\stack_python.py"
pushed item: 1
pushed item: 2
pushed item: 3
pushed item: 4
popped item: 4
stack after popping an element: ['1', '2', '3']
PS C:\Users\lenovo>

```