

PRAKTIKUM 6 – STRUKTUR DATA

NAMA : PRIVINDO FEBRIAN DWI N

NIM : G.211.22.0081

KELAS : TI – B PAGI

1. Implementasi Queue dengan deque

```
from collections import deque

# Inisialisasi deque dengan beberapa elemen
queue = deque(["Ram", "Tarun", "Asif", "John"])
print(queue)

# Menambahkan elemen ke dalam deque
queue.append("Akbar")
print(queue)

queue.append("Birbal")
print(queue)

# Menghapus elemen dari deque menggunakan popleft()
print(queue.popleft())
print(queue.popleft())

print(queue)
```

Algoritma:

- Menggunakan deque dari modul collections untuk mengimplementasikan antrian.
- Menambahkan beberapa elemen ke dalam deque menggunakan append.
- Menghapus elemen pertama dari deque menggunakan popleft.
- Mencetak isi deque setelah beberapa operasi.

2. Tower of Hanoi

```
def TowerOfHanoi(n, source_rod, destination_rod, auxiliary_rod):  
    if n == 0:  
        return  
    TowerOfHanoi(n-1, source_rod, auxiliary_rod, destination_rod)  
    print("Pindah disk", n, "dari", source_rod, "ke", destination_rod)  
    TowerOfHanoi(n-1, auxiliary_rod, source_rod, destination_rod)  
  
# Driver code  
N = 3  
TowerOfHanoi(N, 'A', 'C', 'B')
```

Algoritma:

- Mengimplementasikan rekursi untuk menyelesaikan masalah Tower of Hanoi.
- Fungsi TowerOfHanoi dipanggil secara rekursif untuk memindahkan n disk dari satu tiang ke tiang lain dengan menggunakan tiang bantuan.
- Pesan cetak digunakan untuk menunjukkan langkah-langkah pemindahan disk.

3. Sort Stack using Another Stack

```
def sortStack(input):
    tmpStack = []
    while len(input) > 0:
        tmp = input[-1]
        input.pop()
        while len(tmpStack) > 0 and tmpStack[-1] < tmp:
            input.append(tmpStack[-1])
            tmpStack.pop()
        tmpStack.append(tmp)
    return tmpStack
```

```
def sortArrayUsingStacks(arr, n):
    input = []
    i = 0
    while i < n:
        input.append(arr[i])
        i = i + 1
```

```
    tmpStack = sortStack(input)
    i = 0
    while i < n:
        arr[i] = tmpStack[-1]
        tmpStack.pop()
        i = i + 1
```

```
    return arr
```

```
# Driver code
arr = [10, 5, 15, 45]
n = len(arr)
arr = sortArrayUsingStacks(arr, n)
i = 0
while i < n:
    print(arr[i], end=" ")
    i = i + 1
```

Algoritma:

- Menerapkan algoritma pengurutan tumpukan menggunakan tumpukan tambahan (tmpStack).
- Menggunakan dua loop while: loop pertama untuk memproses elemen dari input ke tmpStack, loop kedua untuk membalikkan elemen-elemen dari tmpStack ke input setelah melakukan pengurutan.
- Fungsi sortArrayUsingStacks mengambil array sebagai input, mengonversinya menjadi tumpukan, mengurutkan tumpukan, dan kemudian mengembalikan array yang diurutkan.