K.R. MANGALAM UNIVERSITY

THE COMPLETE WORLD OF EDUCATION

# ACADEMIC SESSION 2025-26

## Odd Semester

# CIRCULAR LINK LIST OPERATION


**Submitted to:**

**Dr. SWATI**

**Assistant Professor**

**SOED**

**Submitted by:**

**PRIYANSHI**

**2401420029**

**B. Tech (2024-28)**

```
class Node:
    def __init__(self, data):
        self.data = data
        self.next = None


class CircularLinkedList:
    def __init__(self):
        self.head = None

    # Insert at end
    def insert(self, data):
        new_node = Node(data)

        if self.head is None:
            self.head = new_node
            new_node.next = new_node      # circular link
            return

        cur = self.head
        while cur.next != self.head:
            cur = cur.next

        cur.next = new_node
        new_node.next = self.head

    # Delete a node
```

```
def delete(self, key):
    if self.head is None:
        print("List is empty")
        return

    cur = self.head
    prev = None

    # Case: deleting head
    if cur.data == key:
        while cur.next != self.head:
            cur = cur.next
        cur.next = self.head.next
        self.head = self.head.next
        return

    cur = self.head
    while cur.next != self.head:
        prev = cur
        cur = cur.next
        if cur.data == key:
            prev.next = cur.next
            return

    print("Value not found")

# Search element
```

```
def search(self, key):
    if self.head is None:
        return False

    cur = self.head
    while True:
        if cur.data == key:
            return True
        cur = cur.next
        if cur == self.head:
            break
    return False

# Display list
def display(self):
    if self.head is None:
        print("List is empty")
        return

    cur = self.head
    while True:
        print(cur.data, end=" -> ")
        cur = cur.next
        if cur == self.head:
            break
    print("(back to head)")
```

```
# ----------------------------
# Example Usage
# ----------------------------
cll = CircularLinkedList()
cll.insert(10)
cll.insert(20)
cll.insert(30)

print("Circular Linked List:")
cll.display()

print("Search 20:", cll.search(20))
print("Search 40:", cll.search(40))

print("Deleting 20...")
cll.delete(20)

cll.display()
```