Assignment On

‘ADVANCED DATA STRUCTURES AND ALGORITHMS’ (Assignment-2)

Submitted by:
K. Shiva Kiran
2503B05121(M.tech).
Submitted to:
Dr. Rahul Mishra.

Question: Convert The Patient List to A Circular Linked List for a Round-Robin Check-Up System. Implement Insertion and Deletion.

Code :

```
class Patient:     def __init__(self, name,
age, patient_id):         self.name = name
self.age = age         self.id = patient_id
        self.next = None

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

    # Insert patient at end (circular)      def
insert_patient(self, name, age, patient_id):
new_patient = Patient(name, age, patient_id)

        if self.head is None:
self.head = new_patient
new_patient.next = self.head
else:
            temp = self.head
while temp.next != self.head:
temp = temp.next
temp.next = new_patient
            new_patient.next = self.head

        print("Patient inserted:", name)

    # Delete patient by ID      def
delete_patient(self, patient_id):
if self.head is None:
print("List is empty.")
```

```
            return

        temp = self.head
        prev = None

        # Case 1: Only one node         if temp.id ==
patient_id and temp.next == self.head:
            self.head = None
            print("Only patient deleted. List empty now.")
return

        # Case 2: Head deletion with more
nodes         if temp.id == patient_id:
# Find last node              last = self.head
while last.next != self.head:
                last = last.next

            last.next = self.head.next
self.head = self.head.next
            print("Head patient deleted with ID:",
patient_id)             return

        # Case 3: Delete non-head node
while temp.next != self.head:
            prev = temp
temp = temp.next             if
temp.id == patient_id:
prev.next = temp.next
                print("Patient deleted with ID:", patient_id)
return

        print("Patient not found with ID:", patient_id)

    # Display circular list     def
display(self):         if self.head is
None:             print("No patients
in the list.")
            return

        print("\n--- Round Robin Patient List
---")         temp = self.head         while
True:
            print(f"Name: {temp.name}, Age: {temp.age}, ID:
{temp.id}")             temp = temp.next             if temp ==
self.head:                 break
```

```
# -------------------------
# Example Usage
# -------------------------
cplist = CircularPatientList()

cplist.insert_patient("Shiva", 24, 101)
cplist.insert_patient("Nikitha", 22, 102)
cplist.insert_patient("MC", 25, 103)

cplist.display()

cplist.delete_patient(102)

cplist.display()
```

Output: