Rajalakshmi Engineering College

Name: Tejeshwaran P
Email: 241801292@rajalakshmi.edu.in
Roll no: 241801292
Phone: 6383048813
Branch: REC
Department: l AI & DS AF
Batch: 2028
Degree: B.E - AI & DS

Scan to verify results

NeoColab_REC_CS23231_DATA STRUCTURES

REC_DS using C_Week 4_CY

Attempt : 1
Total Mark : 30
Marks Obtained : 30

Section 1 : Coding

1. Problem Statement

A customer support system is designed to handle incoming requests using a queue. Implement a linked list-based queue where each request is represented by an integer. After processing the requests, remove any duplicate requests to ensure that each request is unique and print the remaining requests.

Input Format

The first line of input consists of an integer N, representing the number of

requests to be enqueued.

The second line consists of N space-separated integers, each representing a request. *Output Format*

requests.

The output prints space-separated integers after removing the duplicate

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 5
2 4 2 7 5
Output: 2 4 7 5

Answer

```
// You are using GCC
#include <stdio.h>
#include <stdlib.h>

struct Node {
   int data;    struct
Node* next;
};

void enqueue(struct Node** rear, int data) {
```

```
    struct Node* newNode = (struct
Node*)malloc(sizeof(struct Node));    newNode-                    >data = data;
    newNode->next = NULL;
    if (*rear) {
        (*rear)->next = newNode;
    }
    *rear = newNode;
}

void removeDuplicates(struct Node* head) {
struct Node *ptr1, *ptr2, *dup;    ptr1 = head;
    while (ptr1 != NULL && ptr1->next != NULL) {
ptr2 = ptr1;
        while (ptr2->next                != NULL) {
if (ptr1->data == ptr2-                  >next->data)
{               dup = ptr2-              >next;
ptr2->next = ptr2-                       >next->next;
free(dup);            }                  else {
              ptr2 = ptr2-               >next;
    }
        }
        ptr1 = ptr1->next;
    }
}

void display(struct Node* head) {    while (head != NULL) {        printf("%d ", head-
>data);
        head = head->next;
    }
}

int main() {    int n, val;
    struct Node *front = NULL, *rear = NULL;

    scanf("%d", &n);    for(int i = 0; i < n; i++) {
scanf("%d", &val);        enqueue(&rear, val);        if (front ==
NULL)
            front = rear;
    }
```

```
removeDuplicates(front);     display(front);
    return 0; }
```

Status : Correct *Marks : 10/10*