# DSA ASSIGNMENT

NAME – KAUSHAL KUMAR

USN – 1AY23CS100

- Q1

## CODE

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#define MAX 10
struct Node {
   char usn[15];
   char name[50];
   char programme[50];
   char phone[15];
   struct Node *next;
};
struct Node *queue[MAX];
int front = -1, rear = -1;
struct Node *head = NULL;
void enqueue() {
   if ((rear + 1) % MAX == front) {
```

```
        printf("Queue overflow! Cannot add more students.\n");
        return;
    }
    struct Node *temp = (struct Node *)malloc(sizeof(struct Node));
    printf("Enter USN: ");
    scanf("%s", temp->usn);
    printf("Enter Name: ");
    scanf("%s", temp->name);
    printf("Enter Programme: ");
    scanf("%s", temp->programme);
    printf("Enter Phone Number: ");
    scanf("%s", temp->phone);
    temp->next = NULL;
    if (front == -1) {
        front = 0;
    }
    rear = (rear + 1) % MAX;
    queue[rear] = temp;
    printf("Student %s added to the registration queue.\n", temp->usn);
}
struct Node *dequeue() {
    if (front == -1) {
        printf("Queue underflow! No students in the queue.\n");
        return NULL;
    }
    struct Node *temp = queue[front];
```

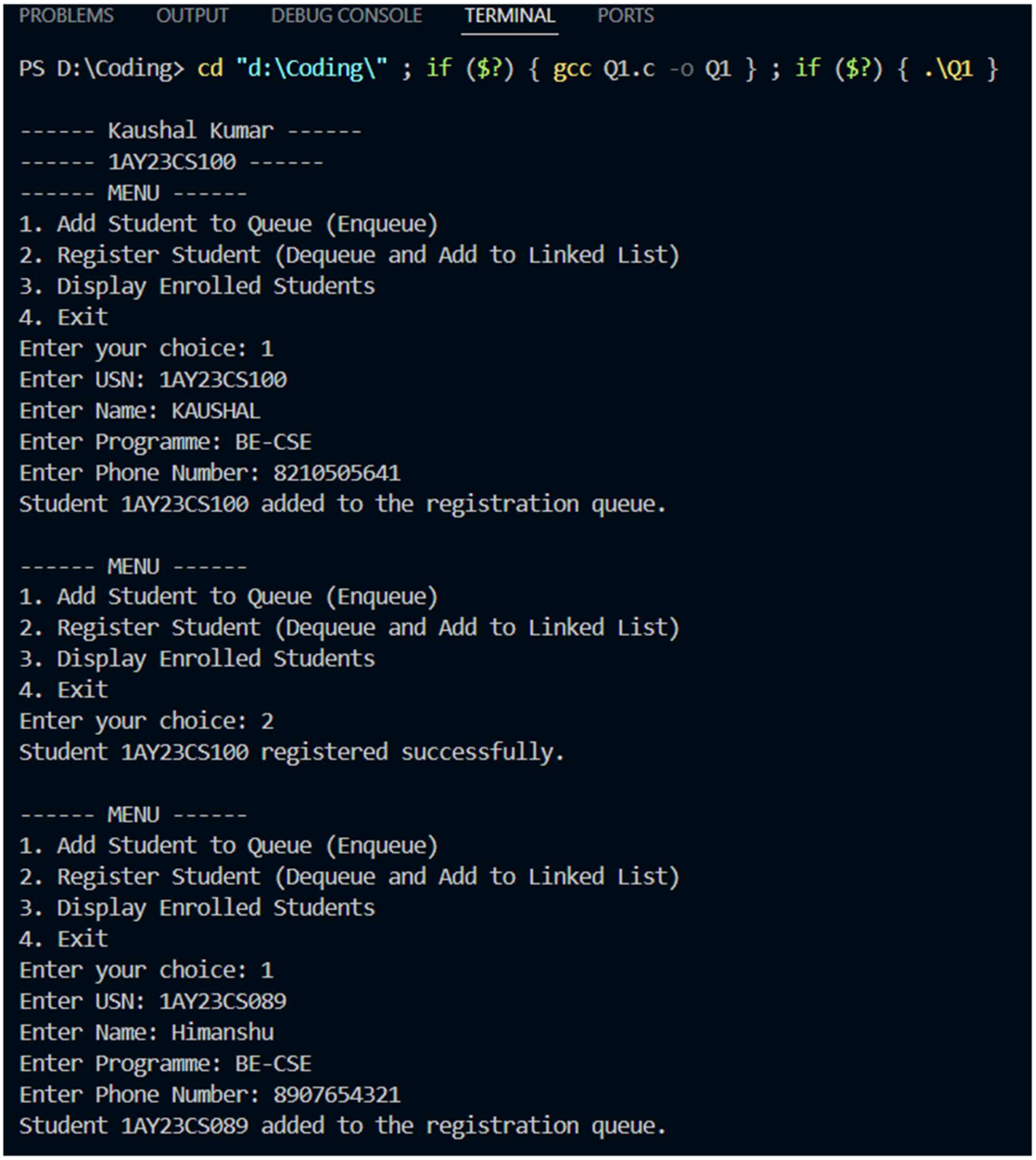
```
    if (front == rear) {
        front = -1;
        rear = -1;
    }
    else {
        front = (front + 1) % MAX;
    }

    return temp;
}
void enqueue_list(struct Node *Node) {
    if (Node == NULL) {
        return;
    }
    Node->next = NULL;
    if (head == NULL) {
        head = Node;
    }
    else {
        struct Node *temp = head;
        while (temp->next != NULL) {
            temp = temp->next;
        }
        temp->next = Node;
    }
    printf("Student %s registered successfully.\n", Node->usn);
```

```
}
void display() {
    if (head == NULL) {
        printf("No students have been enrolled yet.\n");
        return;
    }
    printf("List of enrolled students:\n");
    struct Node *temp = head;
    while (temp != NULL) {
        printf("USN: %s, Name: %s, Programme: %s, Phone: %s\n",
               temp->usn, temp->name, temp->programme, temp->phone);
        temp = temp->next;
    }
}
int main() {
    int choice;
    struct Node *Node;
    printf("\n------ Kaushal Kumar ------");
    printf("\n------ 1AY23CS100 ------");
    while (1) {
        printf("\n------ MENU ------");
        printf("\n1. Add Student to Queue (Enqueue)");
        printf("\n2. Register Student (Dequeue and Add to Linked List)");
        printf("\n3. Display Enrolled Students");
        printf("\n4. Exit");
        printf("\nEnter your choice: ");
```

```
        scanf("%d", &choice);
        switch (choice) {
            case 1:
                enqueue();
                break;
            case 2:
                Node = dequeue();
                if (Node != NULL) {
                    enqueue_list(Node);
                }
                break;
            case 3:
                display();
                break;
            case 4:
                printf("Exiting the system.\n");
                exit(0);
            default:
                printf("Invalid choice! Please try again.\n");
                break;
        }
    }
    return 0;
}
```

# OUTPUT


PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
PS D:\Coding> cd "d:\Coding\" ; if ($?) { gcc Q1.c -o Q1 } ; if ($?) { .\Q1 }
------ Kaushal Kumar ------
------ 1AY23CS100 ------
------ MENU ------
1. Add Student to Queue (Enqueue)
2. Register Student (Dequeue and Add to Linked List)
3. Display Enrolled Students
4. Exit
Enter your choice: 1
Enter USN: 1AY23CS100
Enter Name: KAUSHAL
Enter Programme: BE-CSE
Enter Phone Number: 8210505641
Student 1AY23CS100 added to the registration queue.
------ MENU ------
1. Add Student to Queue (Enqueue)
2. Register Student (Dequeue and Add to Linked List)
3. Display Enrolled Students
4. Exit
Enter your choice: 2
Student 1AY23CS100 registered successfully.
------ MENU ------
1. Add Student to Queue (Enqueue)
2. Register Student (Dequeue and Add to Linked List)
3. Display Enrolled Students
4. Exit
Enter your choice: 1
Enter USN: 1AY23CS089
Enter Name: Himanshu
Enter Programme: BE-CSE
Enter Phone Number: 8907654321
Student 1AY23CS089 added to the registration queue.

```
------ MENU ------
1. Add Student to Queue (Enqueue)
2. Register Student (Dequeue and Add to Linked List)
3. Display Enrolled Students
4. Exit
Enter your choice: 3
List of enrolled students:
USN: 1AY23CS100, Name: KAUSHAL, Programme: BE-CSE, Phone: 8210505641

------ MENU ------
1. Add Student to Queue (Enqueue)
2. Register Student (Dequeue and Add to Linked List)
3. Display Enrolled Students
4. Exit
Enter your choice: 2
Student 1AY23CS089 registered successfully.

------ MENU ------
1. Add Student to Queue (Enqueue)
2. Register Student (Dequeue and Add to Linked List)
3. Display Enrolled Students
4. Exit
Enter your choice: 3
List of enrolled students:
USN: 1AY23CS100, Name: KAUSHAL, Programme: BE-CSE, Phone: 8210505641
USN: 1AY23CS089, Name: Himanshu, Programme: BE-CSE, Phone: 8907654321

------ MENU ------
1. Add Student to Queue (Enqueue)
2. Register Student (Dequeue and Add to Linked List)
3. Display Enrolled Students
4. Exit
Enter your choice: 4
Exiting the system.
PS D:\Coding>
```

- Q2

## CODE

```
#include <stdio.h>
int fibonacci(int n) {
    if (n == 0) {
        return 0;
    }
    else if (n == 1) {
        return 1;
    }
    else {
        return fibonacci(n - 1) + fibonacci(n - 2);
    }
}
int main() {
    int n;
    printf("\n------ Kaushal Kumar ------");
    printf("\n------ 1AY23CS100 ------");
    printf("\nEnter the number of terms: ");
    scanf("%d", &n);
    if (n <= 0) {
        printf("Enter a positive integer.\n");
    }
```

```
    else {
        printf("Fibonacci sequence up to %d terms:\n", n);
        for (int i = 0; i < n; i++) {
            printf("%d ", fibonacci(i));
        }
        printf("\n");
    }

    return 0;
}
```

## OUTPUT