

**NAME: JERRY DAVID R (192424401)**

**COURSE NAME: DATA STRUCTURES FOR MODERN COMPUTING SYSTEMS**

**COURSE CODE: CSA0302**

Experiment 18: Implementation of Queue using Linked List

Code:

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
struct Node {
```

```
    int data;
```

```
    struct Node* next;
```

```
};
```

```
struct Node *front = NULL, *rear = NULL;
```

```
void enqueue() {
```

```
    int value;
```

```
    struct Node* newNode = (struct Node*)malloc(sizeof(struct Node));
```

```
    if(!newNode) {
```

```
        printf("Memory not available\n");
```

```
        return;
```

```
    }
```

```
    printf("Enter value to enqueue: ");
```

```
    scanf("%d", &value);
```

```
    newNode->data = value;
```

```
    newNode->next = NULL;
```

```
    if(rear == NULL) {
```

```
        front = rear = newNode;
```

```
    } else {
```

```
    rear->next = newNode;

    rear = newNode;
}

printf("Value inserted successfully\n");
}
```

```
void dequeue() {
    if(front == NULL) {
        printf("Queue Underflow\n");
        return;
    }

    struct Node* temp = front;
    printf("Deleted element: %d\n", front->data);
    front = front->next;
    if(front == NULL)
        rear = NULL;
    free(temp);
}
```

```
void display() {
    struct Node* temp = front;
    if(front == NULL) {
        printf("Queue is empty\n");
        return;
    }

    printf("Queue elements:\n");
    while(temp != NULL) {
        printf("%d ", temp->data);
        temp = temp->next;
    }

    printf("\n");
}
```

```
}
```

```
int main() {
```

```
    int choice;
```

```
    while(1) {
```

```
        printf("\n--- Queue Menu ---\n");
```

```
        printf("1. Enqueue\n2. Dequeue\n3. Display\n4. Exit\n");
```

```
        printf("Enter your choice: ");
```

```
        scanf("%d", &choice);
```

```
        switch(choice) {
```

```
            case 1: enqueue(); break;
```

```
            case 2: dequeue(); break;
```

```
            case 3: display(); break;
```

```
            case 4: exit(0);
```

```
            default: printf("Invalid choice\n");
```

```
        }
```

```
    }
```

```
}
```

Output:

```
--- Queue Menu ---
1. Enqueue
2. Dequeue
3. Display
4. Exit
Enter your choice: 1
Enter value to enqueue: 20
Value inserted successfully
```

```
--- Queue Menu ---
1. Enqueue
2. Dequeue
3. Display
4. Exit
Enter your choice: 3
Queue elements:
20
```

```
--- Queue Menu ---
1. Enqueue
2. Dequeue
3. Display
4. Exit
Enter your choice: 2
Deleted element: 20
```

```
--- Queue Menu ---
1. Enqueue
2. Dequeue
3. Display
4. Exit
Enter your choice: 4
```