LINEAR QUEUE

```
#include <stdio.h>

int rear = -1, size, front = 0, count = 0;

int main() {
    printf("Enter the size of the queue: ");
    scanf("%d", &size);
    int queue[size];

    while (1) {
        int choice;
        printf("1. Insert \n");
        printf("2. Delete \n");
        printf("3. Display \n");
        printf("4. Exit\n");
        printf("Enter your choice: ");
        scanf("%d", &choice);

        switch (choice) {
            case 1:
                if (count == size) {
                    printf("Queue overflow\n");
                } else {
                    rear++;
                    printf("Enter value to insert: ");
                    scanf("%d", &queue[rear]);
                    count++;
                    printf("Inserted into queue\n");
                }
                break;
            case 2:
                if (count == 0) {
                    printf("Queue underflow\n");
                } else {
                    printf("Deleted from queue:\n", queue[front]);
                    front++;
                    count--;
                }
                break;
            case 3:
                if (count == 0) {
                    printf("Queue is empty\n");
```

```
            } else {
                for (int i = front; i <= rear; i++) {
                    printf("%d ", queue[i]);
                }
                printf("\n");
            }
            break;
        case 4:
            printf("Exit");
            return 0;
        default:
            printf("Invalid choice\n");
            break;
    }
  }
}
```

```
Enter the size of the queue: 2
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 1
Enter value to insert: 2
Inserted into queue
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 1
Enter value to insert: 3
Inserted into queue
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 1
Queue overflow
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 3
2 3
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 2
Deleted from queue:
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 2
Deleted from queue:
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 2
Queue underflow
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 3
Queue is empty
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 4
Exit
Process returned 0 (0x0)   execution time : 28.063 s
Press any key to continue.
```

2) CIRCULAR QUEUE

```
int main() {
    printf("Enter the size of the queue: ");
    scanf("%d", &size);
    int queue[size];

    while (1) {
        int choice;
        printf("1. Insert \n");
        printf("2. Delete \n");
        printf("3. Display \n");
        printf("4. Exit\n");
        printf("Enter your choice: ");
        scanf("%d", &choice);

        switch (choice) {
            case 1:
                if (count == size) {
                    printf("Queue overflow\n");
                } else {
                    rear = (rear + 1) % size;
                    printf("Enter value to insert: ");
                    scanf("%d", &queue[rear]);
                    count++;
                    printf("Inserted into queue\n");
                }
                break;
            case 2:
                if (count == 0) {
                    printf("Queue underflow\n");
                } else {
                    printf("Deleted from queue: %d\n", queue[front]);
                    front = (front + 1) % size;
                    count--;
                }
                break;
            case 3:
                if (count == 0) {
```

```
                printf("Queue is empty\n");
            } else {
                for (int i = 0; i < count; i++) {
                    printf("%d ", queue[(front + i) % size]);
                }
                printf("\n");
            }
            break;
        case 4:
            printf("Exit\n");
            return 0;
        default:
            printf("Invalid choice\n");
            break;
    }
}
```

```
Enter the size of the queue: 2
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 1
Enter value to insert: 2
Inserted into queue
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 1
Enter value to insert: 4
Inserted into queue
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 1
Queue overflow
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 3
2 4
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 2
Deleted from queue: 2
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 2
Deleted from queue: 4
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 2
Queue underflow
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 3
Queue is empty
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 4
Exit

Process returned 0 (0x0)   execution time : 24.156 s
Press any key to continue.
```