Name - Sanjoy Saha

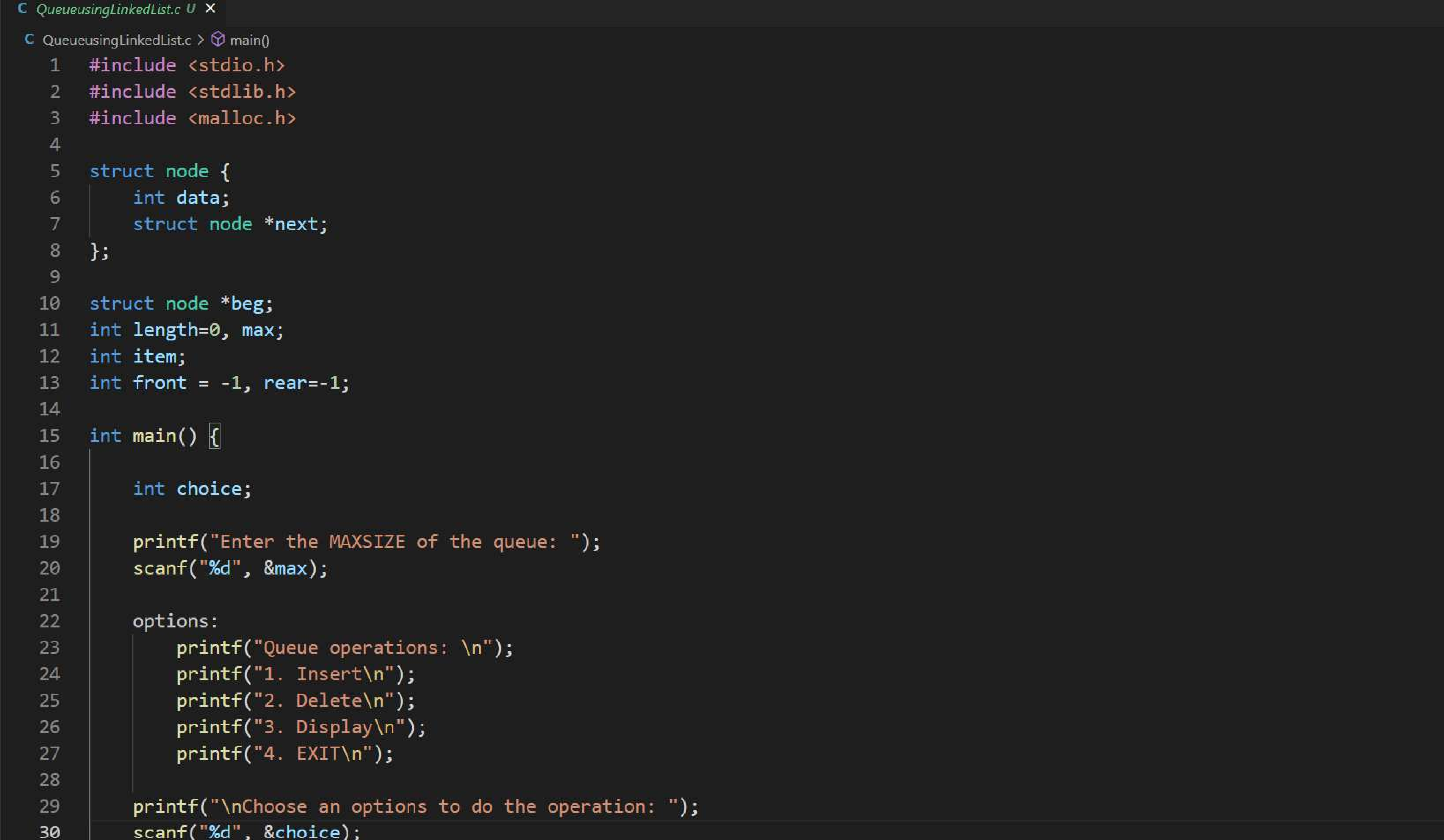
Stream - Computer Science & Engineering

Sec - A

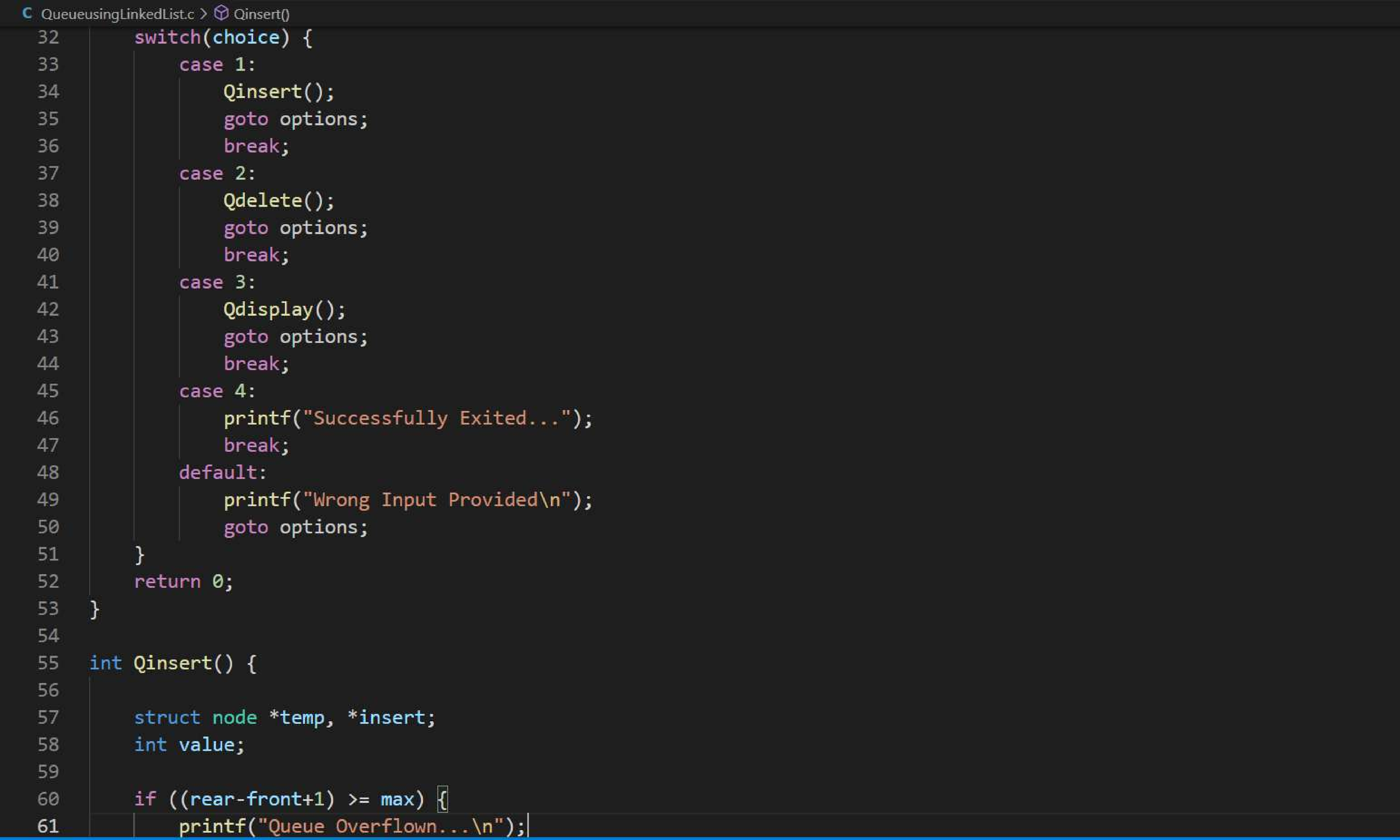
Roll no. - 3

University Roll no. :- 10900120003

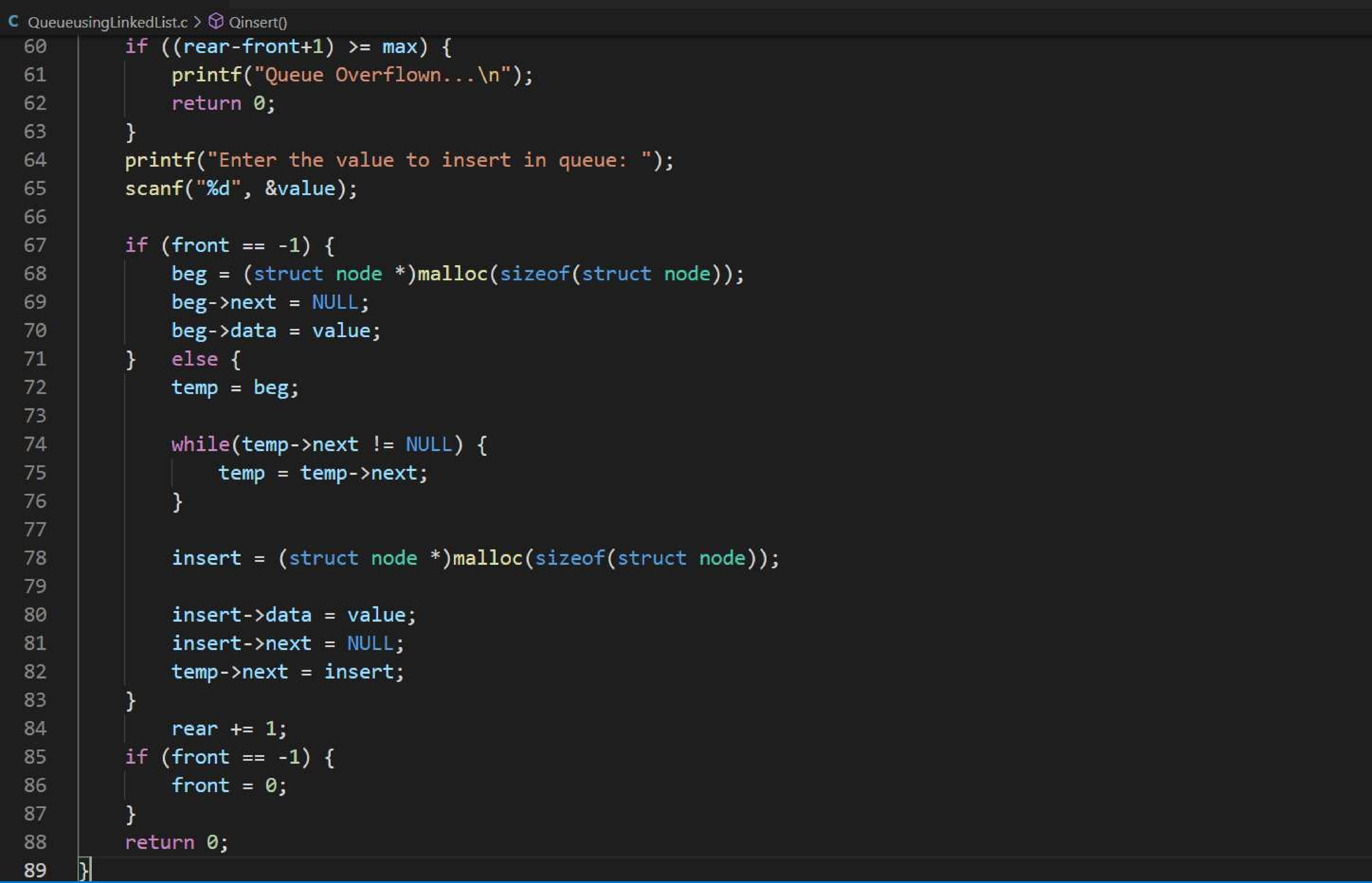
Subject - DSA Lab



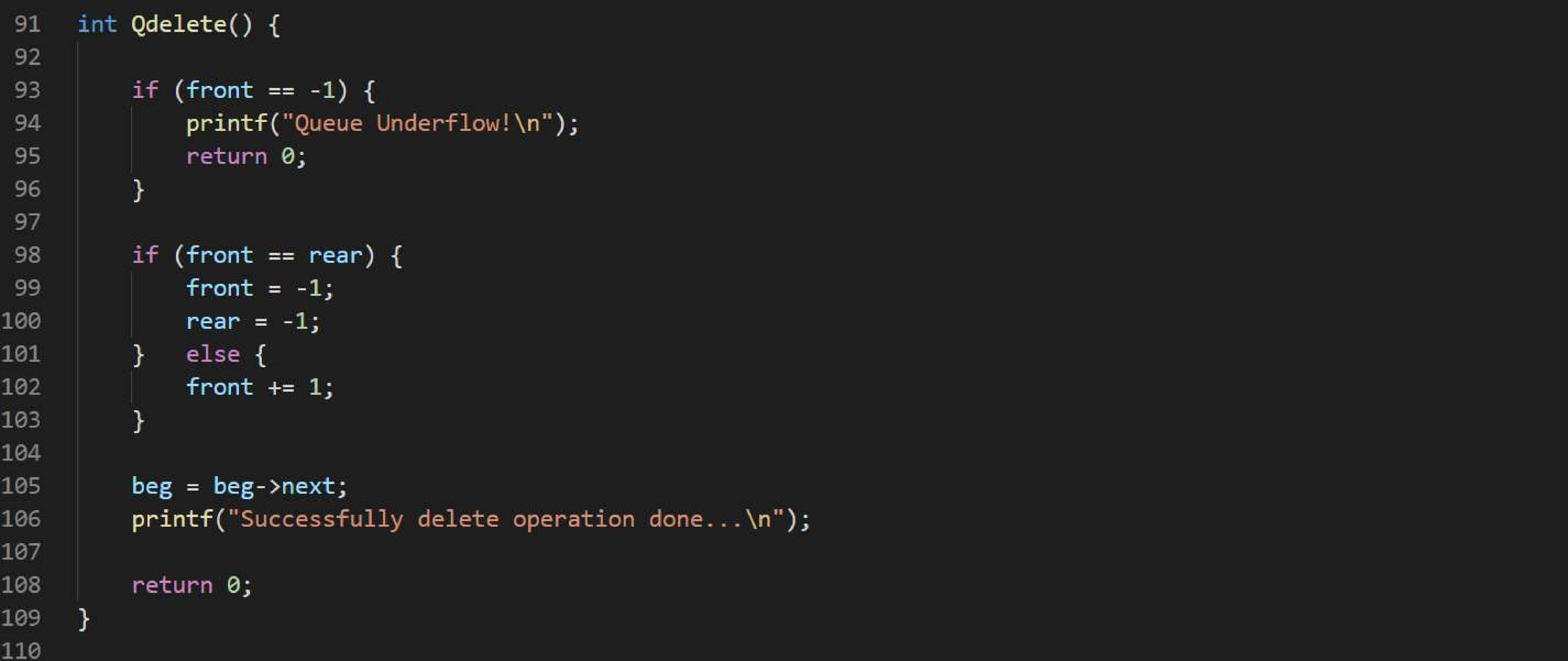
```c
#include <stdio.h>
#include <stdlib.h>
#include <malloc.h>

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

struct node *beg;
int length=0, max;
int item;
int front = -1, rear=-1;

int main() {

    int choice;

    printf("Enter the MAXSIZE of the queue: ");
    scanf("%d", &max);

    options:
        printf("Queue operations: \n");
        printf("1. Insert\n");
        printf("2. Delete\n");
        printf("3. Display\n");
        printf("4. EXIT\n");

    printf("\nChoose an options to do the operation: ");
    scanf("%d", &choice);
```



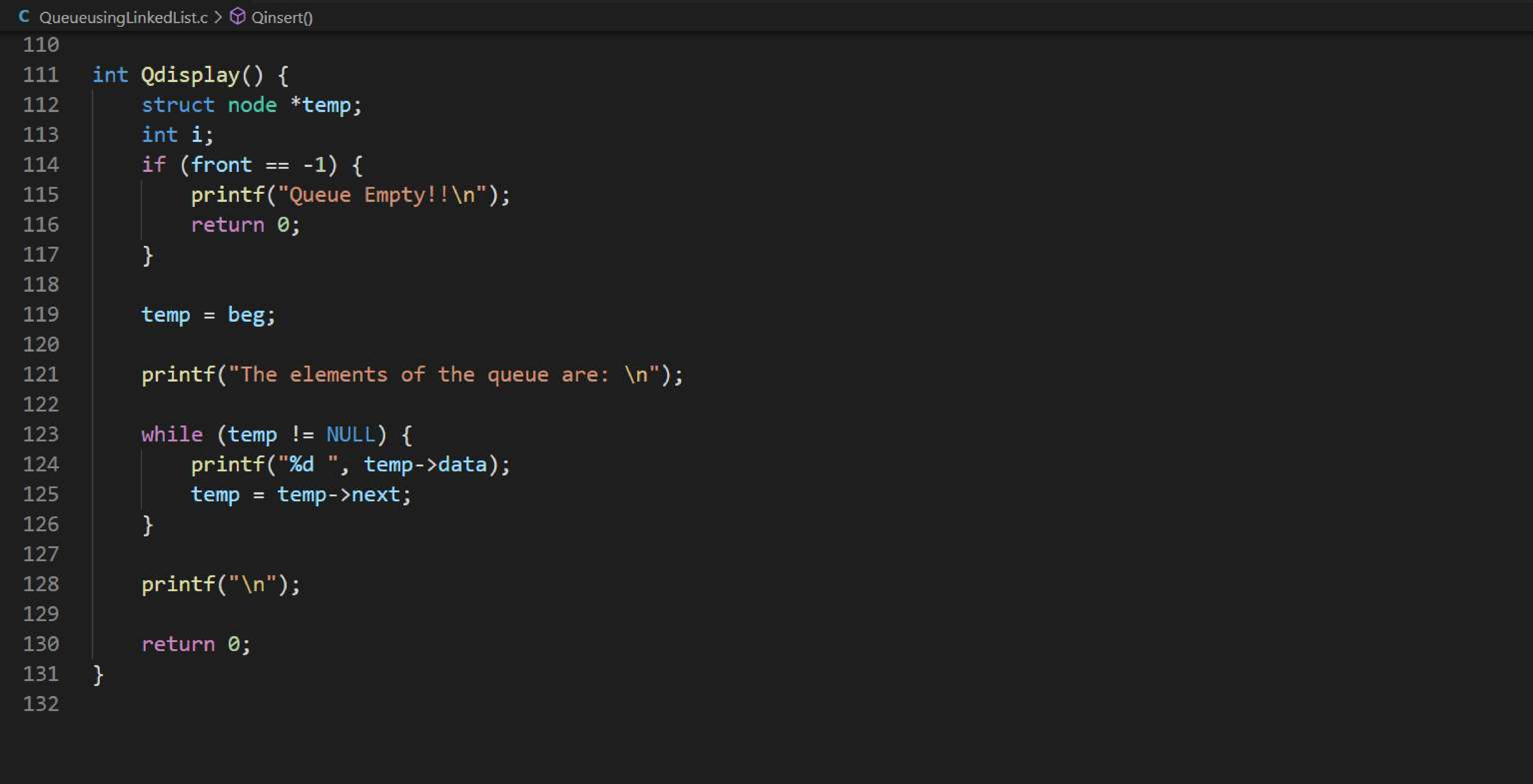
```c
    switch(choice) {
        case 1:
            Qinsert();
            goto options;
            break;
        case 2:
            Qdelete();
            goto options;
            break;
        case 3:
            Qdisplay();
            goto options;
            break;
        case 4:
            printf("Successfully Exited...");
            break;
        default:
            printf("Wrong Input Provided\n");
            goto options;
    }
    return 0;
}

int Qinsert() {

    struct node *temp, *insert;
    int value;

    if ((rear-front+1) >= max) {
        printf("Queue Overflown...\n");
```



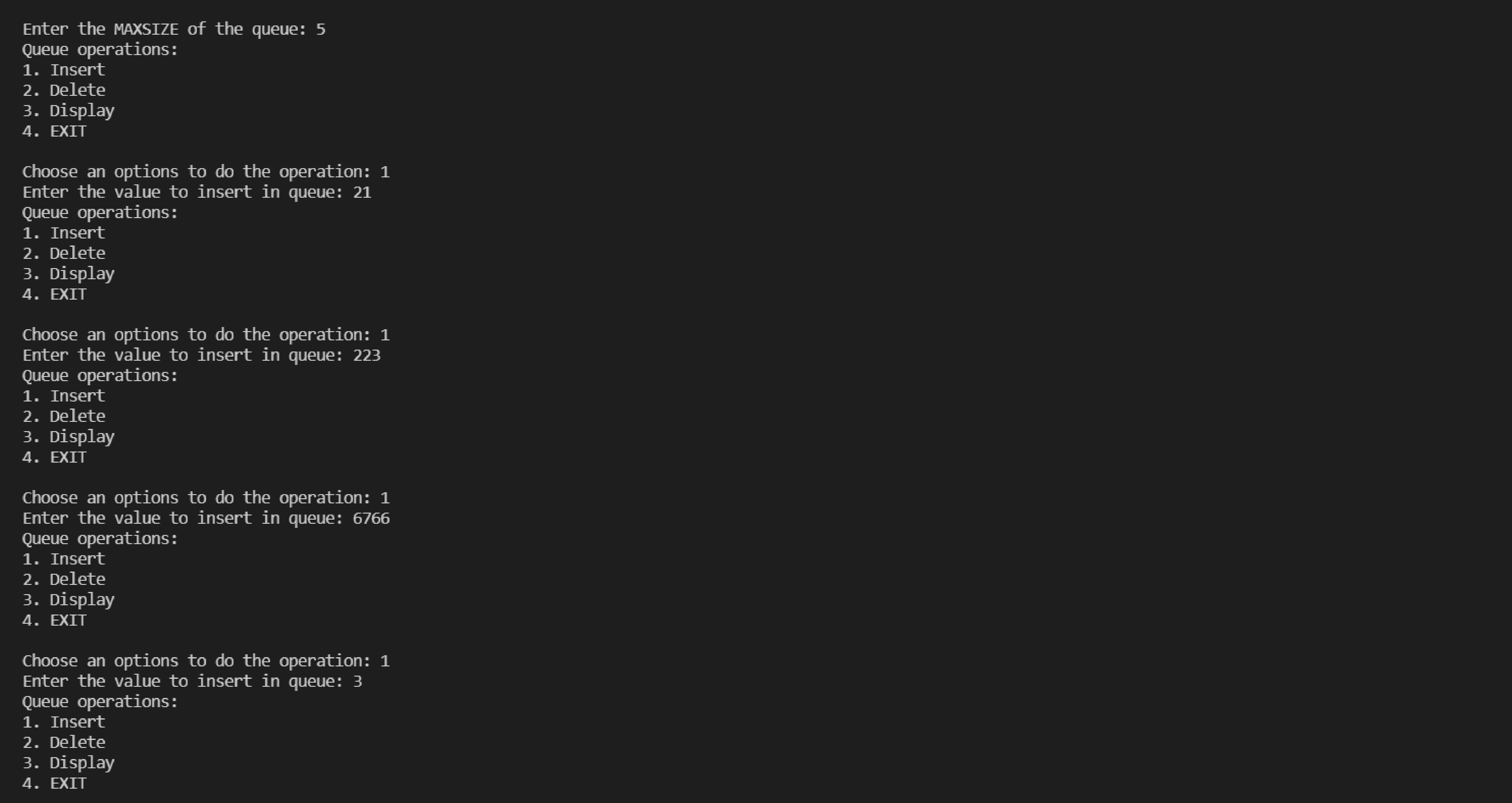
```c
    if ((rear-front+1) >= max) {
        printf("Queue Overflown...\n");
        return 0;
    }
    printf("Enter the value to insert in queue: ");
    scanf("%d", &value);

    if (front == -1) {
        beg = (struct node *)malloc(sizeof(struct node));
        beg->next = NULL;
        beg->data = value;
    }   else {
        temp = beg;

        while(temp->next != NULL) {
            temp = temp->next;
        }

        insert = (struct node *)malloc(sizeof(struct node));

        insert->data = value;
        insert->next = NULL;
        temp->next = insert;
    }
        rear += 1;
    if (front == -1) {
        front = 0;
    }
    return 0;
}
```



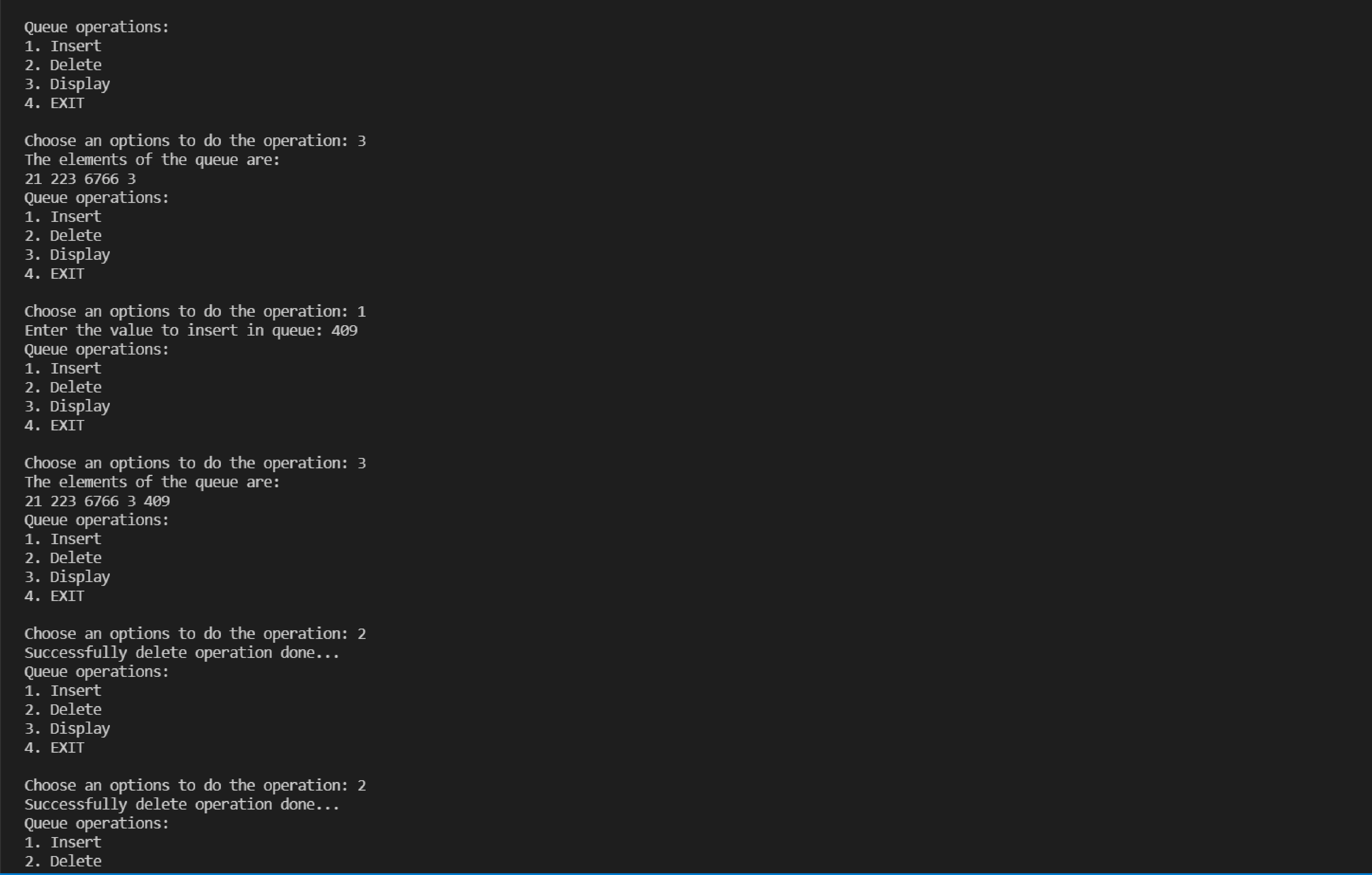
```c
int Qdelete() {

    if (front == -1) {
        printf("Queue Underflow!\n");
        return 0;
    }

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

    beg = beg->next;
    printf("Successfully delete operation done...\n");

    return 0;
}

```



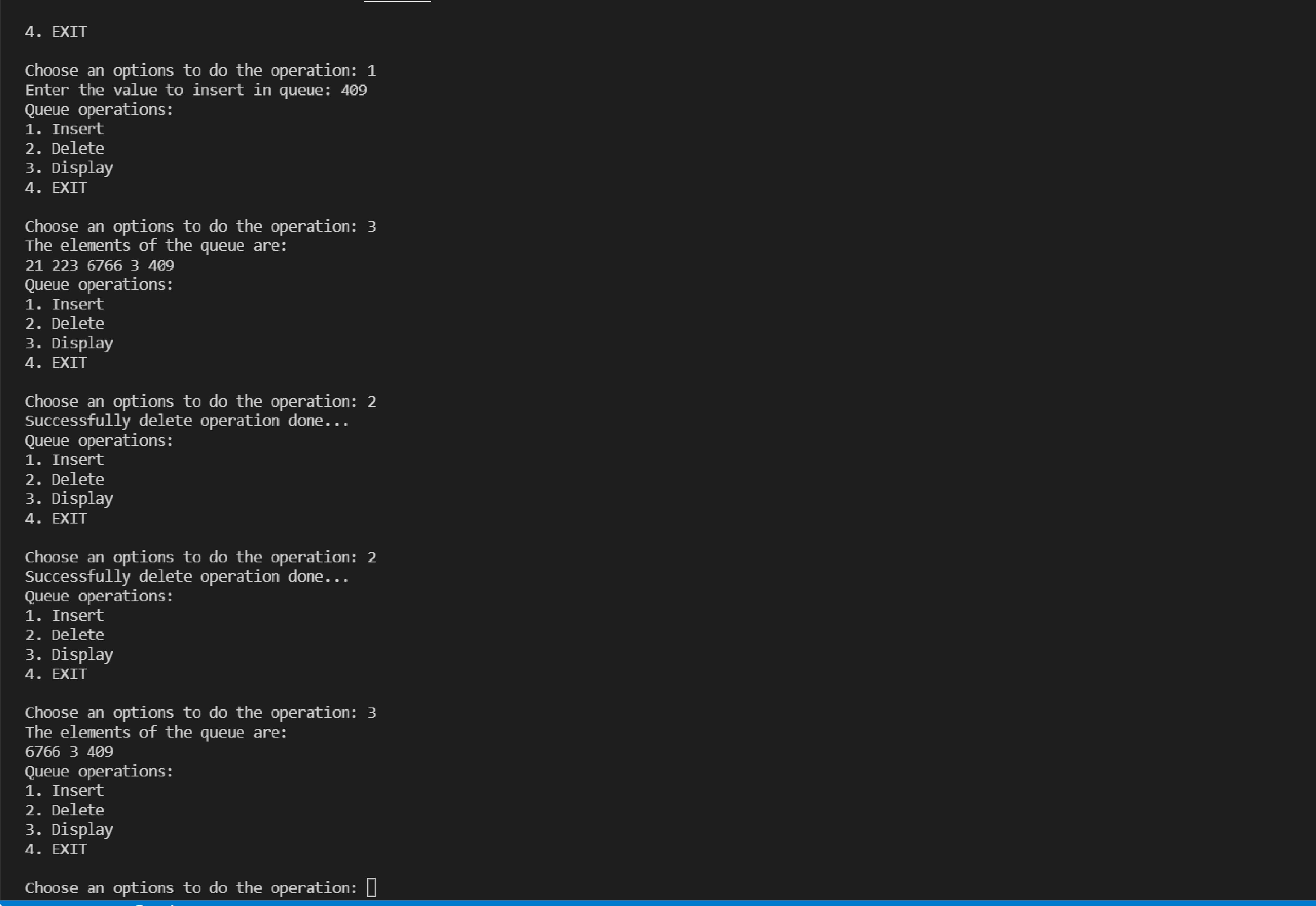
```c

int Qdisplay() {
    struct node *temp;
    int i;
    if (front == -1) {
        printf("Queue Empty!!\n");
        return 0;
    }

    temp = beg;

    printf("The elements of the queue are: \n");

    while (temp != NULL) {
        printf("%d ", temp->data);
        temp = temp->next;
    }

    printf("\n");

    return 0;
}

```



```
Enter the MAXSIZE of the queue: 5
Queue operations:
1. Insert
2. Delete
3. Display
4. EXIT

Choose an options to do the operation: 1
Enter the value to insert in queue: 21
Queue operations:
1. Insert
2. Delete
3. Display
4. EXIT

Choose an options to do the operation: 1
Enter the value to insert in queue: 223
Queue operations:
1. Insert
2. Delete
3. Display
4. EXIT

Choose an options to do the operation: 1
Enter the value to insert in queue: 6766
Queue operations:
1. Insert
2. Delete
3. Display
4. EXIT

Choose an options to do the operation: 1
Enter the value to insert in queue: 3
Queue operations:
1. Insert
2. Delete
3. Display
4. EXIT
```



```
Queue operations:
1. Insert
2. Delete
3. Display
4. EXIT

Choose an options to do the operation: 3
The elements of the queue are:
21 223 6766 3
Queue operations:
1. Insert
2. Delete
3. Display
4. EXIT

Choose an options to do the operation: 1
Enter the value to insert in queue: 409
Queue operations:
1. Insert
2. Delete
3. Display
4. EXIT

Choose an options to do the operation: 3
The elements of the queue are:
21 223 6766 3 409
Queue operations:
1. Insert
2. Delete
3. Display
4. EXIT

Choose an options to do the operation: 2
Successfully delete operation done...
Queue operations:
1. Insert
2. Delete
3. Display
4. EXIT

Choose an options to do the operation: 2
Successfully delete operation done...
Queue operations:
1. Insert
2. Delete
```



```
4. EXIT

Choose an options to do the operation: 1
Enter the value to insert in queue: 409
Queue operations:
1. Insert
2. Delete
3. Display
4. EXIT

Choose an options to do the operation: 3
The elements of the queue are:
21 223 6766 3 409
Queue operations:
1. Insert
2. Delete
3. Display
4. EXIT

Choose an options to do the operation: 2
Successfully delete operation done...
Queue operations:
1. Insert
2. Delete
3. Display
4. EXIT

Choose an options to do the operation: 2
Successfully delete operation done...
Queue operations:
1. Insert
2. Delete
3. Display
4. EXIT

Choose an options to do the operation: 3
The elements of the queue are:
6766 3 409
Queue operations:
1. Insert
2. Delete
3. Display
4. EXIT

Choose an options to do the operation: 
```