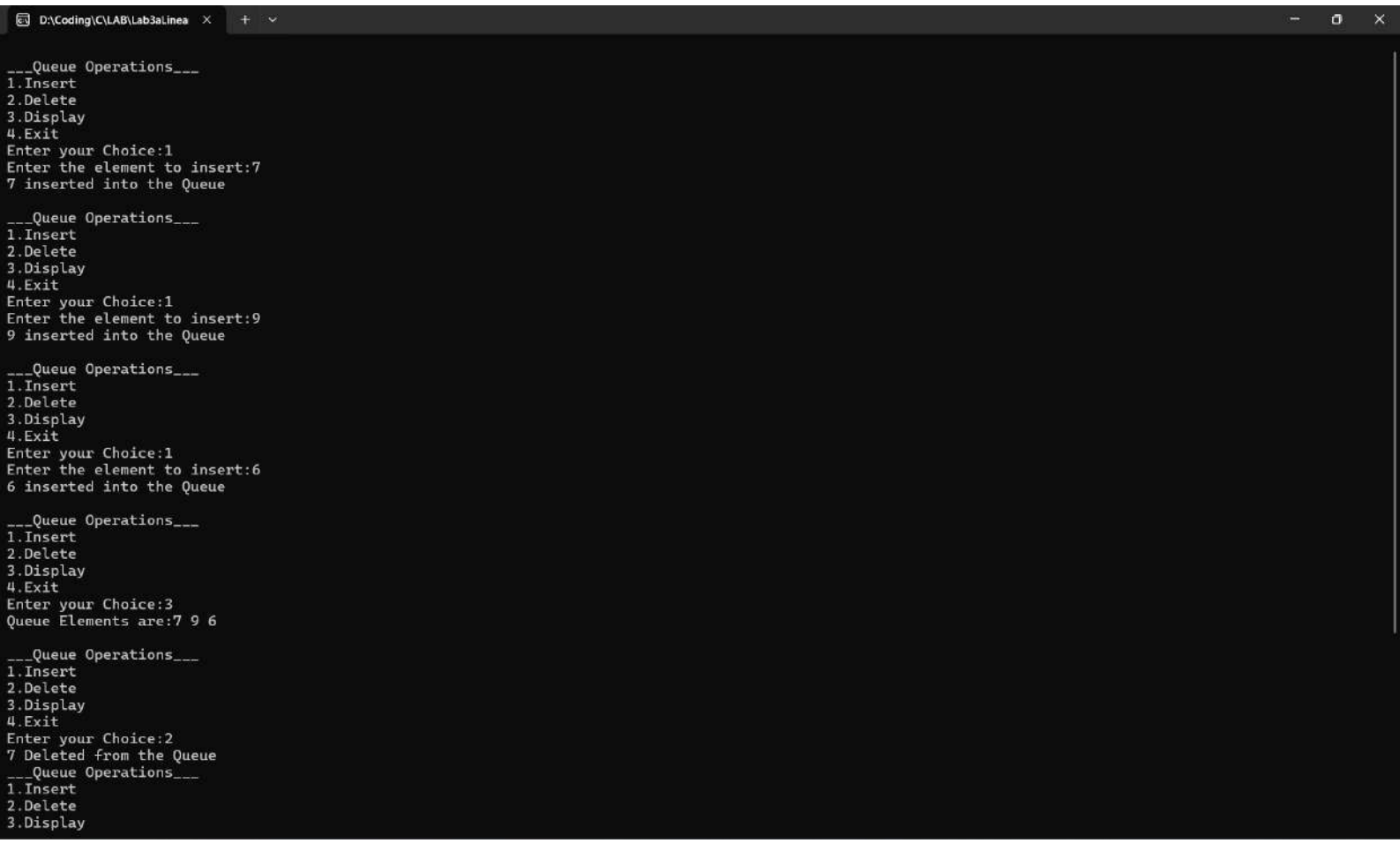
Start here | Lab1stack.c | Lab3aLinearqueue.c

```c
#include<stdio.h>
# define MAX 5
int front=-1,rear=-1;
int queue[MAX];
void enqueue(){
    int value;
    if(rear==MAX-1)
        printf("Queue Overflow! cannot insert into Queue");
    else if(front==-1){
        front=rear=0;
        printf("Enter the element to insert:");
        scanf("%d",&value);
        queue[rear]=value;
        printf("%d inserted into the Queue\n",value);
    }
    else{
        printf("Enter the element to insert:");
        scanf("%d",&value);
        rear++;
        queue[rear]=value;
        printf("%d inserted into the Queue\n",value);

    }
}

void dequeue(){
    if(front==-1 && rear==-1)
        printf("Queue Underflow! No Elements to delete.\n");
    else if(front==rear)
        front=rear=-1;
    else{
        printf("%d Deleted from the Queue\n",queue[front]);
        front++;
    }
}

void display(){
    if(front==-1)
        printf("Queue is empty\n");
    else{
```

Start here | Lab1stack.c | Lab3aLinearqueue.c

```c
    else{
        printf("%d Deleted from the Queue\n",queue[front]);
        front++;
    }
}

void display(){
    if(front==-1)
        printf("Queue is empty\n");
    else{
        printf("Queue Elements are:");
        for(int i=front;i<=rear;i++){
            printf("%d ",queue[i]);
        }
        printf("\n");
    }
}

int main(){
    int choice;
    while(1){
        printf("\n___Queue Operations___\n");
        printf("1.Insert\n2.Delete\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:printf("Exiting program!\n");
                   return 0;
            default:printf("Invalid choice!\n");
        }
    }
    return 0;
}
```

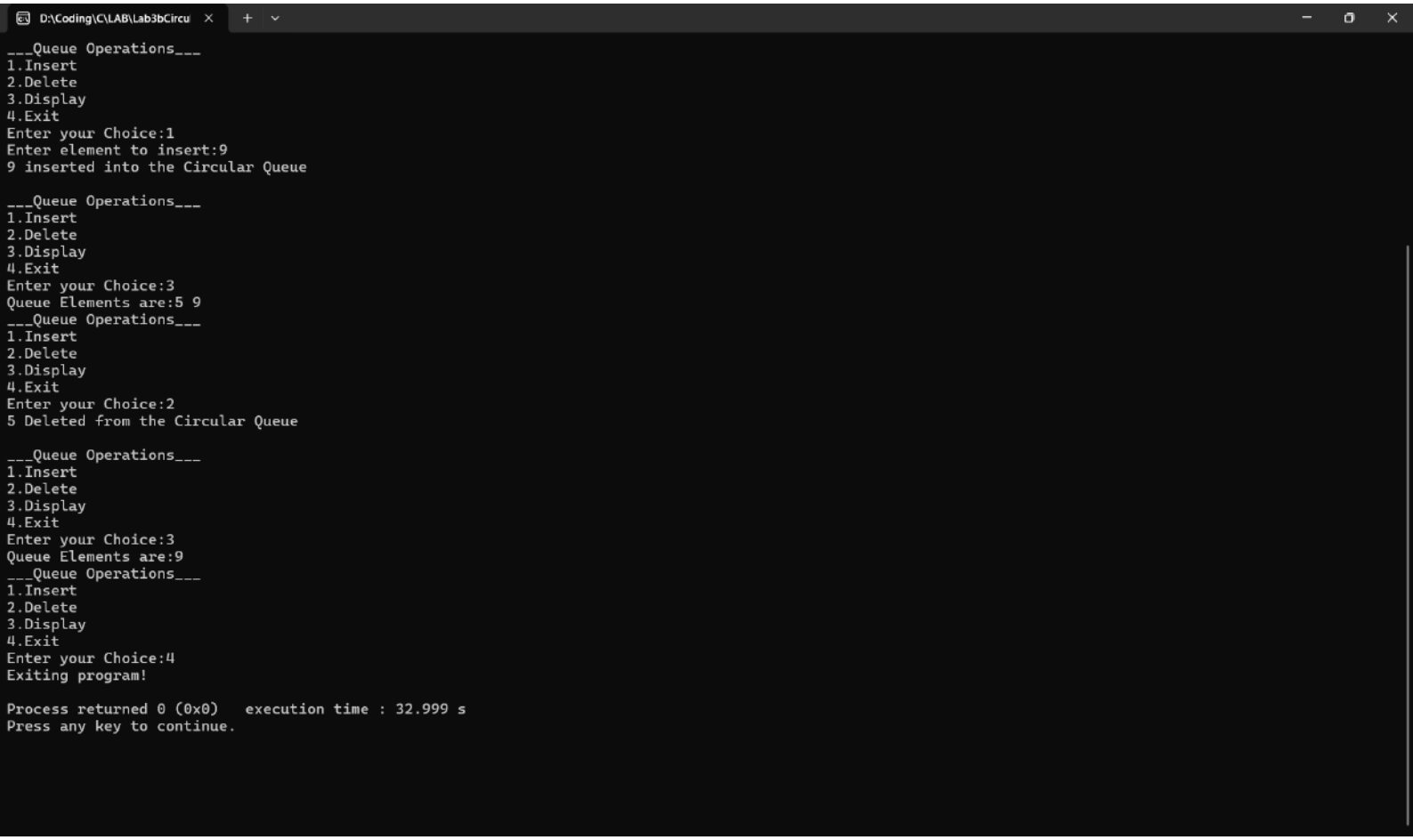

D:\Coding\C\LAB\Lab3aLinea
___Queue Operations___
1.Insert
2.Delete
3.Display
4.Exit
Enter your Choice:1
Enter the element to insert:7
7 inserted into the Queue
___Queue Operations___
1.Insert
2.Delete
3.Display
4.Exit
Enter your Choice:1
Enter the element to insert:9
9 inserted into the Queue
___Queue Operations___
1.Insert
2.Delete
3.Display
4.Exit
Enter your Choice:1
Enter the element to insert:6
6 inserted into the Queue
___Queue Operations___
1.Insert
2.Delete
3.Display
4.Exit
Enter your Choice:3
Queue Elements are:7 9 6
___Queue Operations___
1.Insert
2.Delete
3.Display
4.Exit
Enter your Choice:2
7 Deleted from the Queue
___Queue Operations___
1.Insert
2.Delete
3.Display

Start here | Lab1stack.c | Lab3aLinearqueue.c | Lab3bCircularqueue.c

```c
#include<stdio.h>
#define MAX 5
int queue[MAX];
int front=-1,rear=-1;

void enqueue(){
    int item;
    printf("Enter element to insert:");
    scanf("%d",&item);
    if(((rear+1)%MAX)==front){
        printf("Queue Overflow!");
        return ;
    }
    else if(front==-1 && rear==-1)
        front=rear=0;
    else
        rear=(rear+1)%MAX;
    queue[rear]=item;
    printf("%d inserted into the Circular Queue\n",item);
}

void dequeue(){
    if(front==-1 && rear==-1){
        printf("Queue Underflow! No Elements to delete.\n");
        return ;
    }
    printf("%d Deleted from the Circular Queue\n",queue[front]);
    if(front==rear)
        front=rear=-1;
    else
        front=(front+1)%MAX;
}

void display(){
    int i=front;
    if(front==-1 && rear==-1)
        printf("Queue is empty\n");
```

Start here | Lab1stack.c | Lab3aLinearqueue.c | Lab3bCircularqueue.c

```c
void display(){
    int i=front;
    if(front==-1 && rear==-1)
        printf("Queue is empty\n");
    else{
        printf("Queue Elements are:");
        while(i!=rear){
            printf("%d ",queue[i]);
            i=(i+1)%MAX;
        }
        printf("%d",queue[rear]);
    }
}

int main(){
    int choice;
    while(1){
        printf("\n___Queue Operations___\n");
        printf("1.Insert\n2.Delete\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:printf("Exiting program!\n");
                   return 0;
            default:printf("Invalid choice!\n");
        }
    }
    return 0;
}
```


D:\Coding\C\LAB\Lab3bCircu
___Queue Operations___
1.Insert
2.Delete
3.Display
4.Exit
Enter your Choice:1
Enter element to insert:9
9 inserted into the Circular Queue
___Queue Operations___
1.Insert
2.Delete
3.Display
4.Exit
Enter your Choice:3
Queue Elements are:5 9
___Queue Operations___
1.Insert
2.Delete
3.Display
4.Exit
Enter your Choice:2
5 Deleted from the Circular Queue
___Queue Operations___
1.Insert
2.Delete
3.Display
4.Exit
Enter your Choice:3
Queue Elements are:9
___Queue Operations___
1.Insert
2.Delete
3.Display
4.Exit
Enter your Choice:4
Exiting program!
Process returned 0 (0x0) execution time : 32.999 s
Press any key to continue.