Program - 09

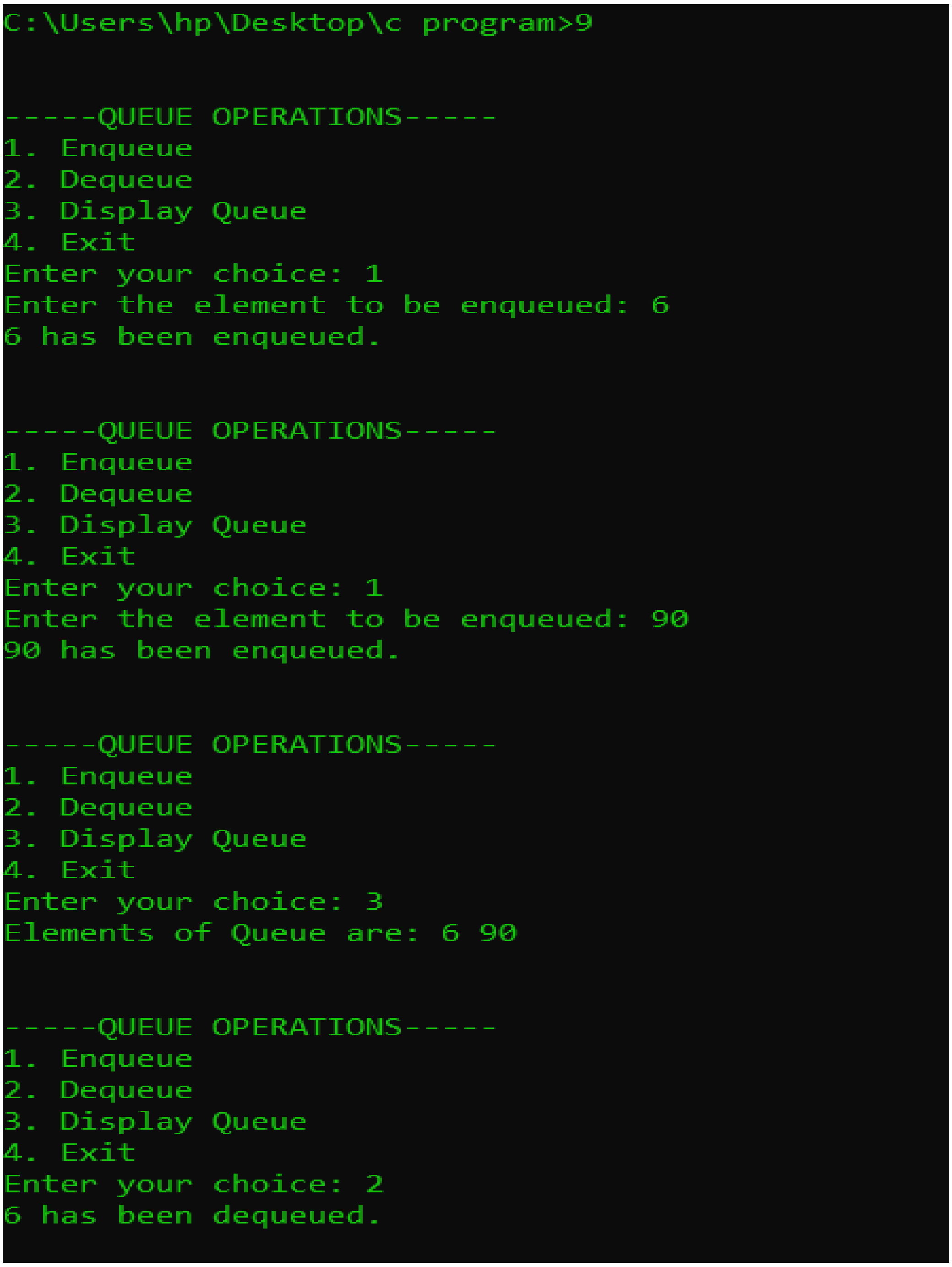
Title : **Write a Program to implement the operations of Queue using linked list**

Code:

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

// Define the Queue node structure
struct QueueNode {
   int data;
   struct QueueNode *next;
};

// Define the Queue structure
struct Queue {
   struct QueueNode *front;
   struct QueueNode *rear;
};

// Function to create a new Queue node
struct QueueNode* createNode(int data) {
   struct QueueNode *newNode = (struct QueueNode*) malloc(sizeof(struct QueueNode));
   newNode->data = data;
   newNode->next = NULL;
   return newNode;
}

// Function to check if the Queue is empty
int isEmpty(struct Queue *queue) {
   return (queue->front == NULL);
}

// Function to add an element to the Queue
void enqueue(struct Queue *queue, int data) {
   struct QueueNode *newNode = createNode(data);
   if(isEmpty(queue)) {
      queue->front = newNode;
      queue->rear = newNode;
   } else {
      queue->rear->next = newNode;
      queue->rear = newNode;
   }
   printf("%d has been enqueued.\n", data);
}

// Function to remove an element from the Queue
void dequeue(struct Queue *queue) {
   if(isEmpty(queue)) {
      printf("Queue is empty.\n");
   } else {
      struct QueueNode *temp = queue->front;
      queue->front = queue->front->next;
      printf("%d has been dequeued.\n", temp->data);
      free(temp);
   }
}
```

```
// Function to display the elements of the Queue
void display(struct Queue *queue) {
    if(isEmpty(queue)) {
        printf("Queue is empty.\n");
    } else {
        printf("Elements of Queue are: ");
        struct QueueNode *temp = queue->front;
        while(temp != NULL) {
            printf("%d ", temp->data);
            temp = temp->next;
        }
        printf("\n");
    }
}

int main() {
    struct Queue *queue = (struct Queue*) malloc(sizeof(struct Queue));
    queue->front = NULL;
    queue->rear = NULL;

    int choice, data;

    do {
        printf("\n\n-----QUEUE OPERATIONS-----\n");
        printf("1. Enqueue\n");
        printf("2. Dequeue\n");
        printf("3. Display Queue\n");
        printf("4. Exit\n");
        printf("Enter your choice: ");
        scanf("%d", &choice);

        switch(choice) {
            case 1:
                printf("Enter the element to be enqueued: ");
                scanf("%d", &data);
                enqueue(queue, data);
                break;
            case 2:
                dequeue(queue);
                break;
            case 3:
                display(queue);
                break;
            case 4:
                printf("Exiting...\n");
                exit(0);
            default:
                printf("Invalid choice.\n");
        }
    } while(1);

    return 0;
}
```

Output:



```
C:\Users\hp\Desktop\c program>9


-----QUEUE OPERATIONS-----
1. Enqueue
2. Dequeue
3. Display Queue
4. Exit
Enter your choice: 1
Enter the element to be enqueued: 6
6 has been enqueued.


-----QUEUE OPERATIONS-----
1. Enqueue
2. Dequeue
3. Display Queue
4. Exit
Enter your choice: 1
Enter the element to be enqueued: 90
90 has been enqueued.


-----QUEUE OPERATIONS-----
1. Enqueue
2. Dequeue
3. Display Queue
4. Exit
Enter your choice: 3
Elements of Queue are: 6 90


-----QUEUE OPERATIONS-----
1. Enqueue
2. Dequeue
3. Display Queue
4. Exit
Enter your choice: 2
6 has been dequeued.
```

Date : ___/____/______	Teacher Sign ....