3. a) WAP to simulate the working of a queue of integers using an array. Provide the following operations: Insert, Delete, Display

The programshould print appropriate messagesfor queue empty and queue overflow conditions

```
#include <stdio.h>

#define MAX 5   // Maximum size of queue

int queue[MAX];
int front = -1, rear = -1;

// Function to insert an element into the queue
void insert() {
   int value;
   if (rear == MAX - 1) {
      printf("Queue Overflow! Cannot insert element.\n");
   } else {
      if (front == -1)  // First insertion
         front = 0;
      printf("Enter the value to insert: ");
      scanf("%d", &value);
      rear++;
      queue[rear] = value;
      printf("%d inserted into queue.\n", value);
   }
}
```

```
// Function to delete an element from the queue
void delete() {
    if (front == -1 || front > rear) {
        printf("Queue is Empty! Cannot delete element.\n");
    } else {
        printf("%d deleted from queue.\n", queue[front]);
        front++;
    }
}

// Function to display the queue
void display() {
    if (front == -1 || front > rear) {
        printf("Queue is Empty.\n");
    } else {
        printf("Queue elements are:\n");
        for (int i = front; i <= rear; i++) {
            printf("%d ", queue[i]);
        }
        printf("\n");
    }
}

int main() {
    int choice;

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

```
        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:
                insert();
                break;
            case 2:
                delete();
                break;
            case 3:
                display();
                break;
            case 4:
                printf("Exiting program.\n");
                break;
            default:
                printf("Invalid choice! Try again.\n");
        }
    } while (choice != 4);

    return 0;
}
```

OUTPUT:

```
--- Queue Menu ---
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 1
Enter the value to insert: 10
10 inserted into queue.

--- Queue Menu ---
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 1
Enter the value to insert: 12
12 inserted into queue.

--- Queue Menu ---
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 1
Enter the value to insert: 13
13 inserted into queue.

--- Queue Menu ---
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 1
Enter the value to insert: 14
14 inserted into queue.

--- Queue Menu ---
1. Insert
2. Delete
```

```
Enter the value to insert: 14
14 inserted into queue.

--- Queue Menu ---
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 1
Enter the value to insert: 15
15 inserted into queue.

--- Queue Menu ---
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 2
10 deleted from queue.

--- Queue Menu ---
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 3
Queue elements are:
12 13 14 15

--- Queue Menu ---
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 4
Exiting program.
```