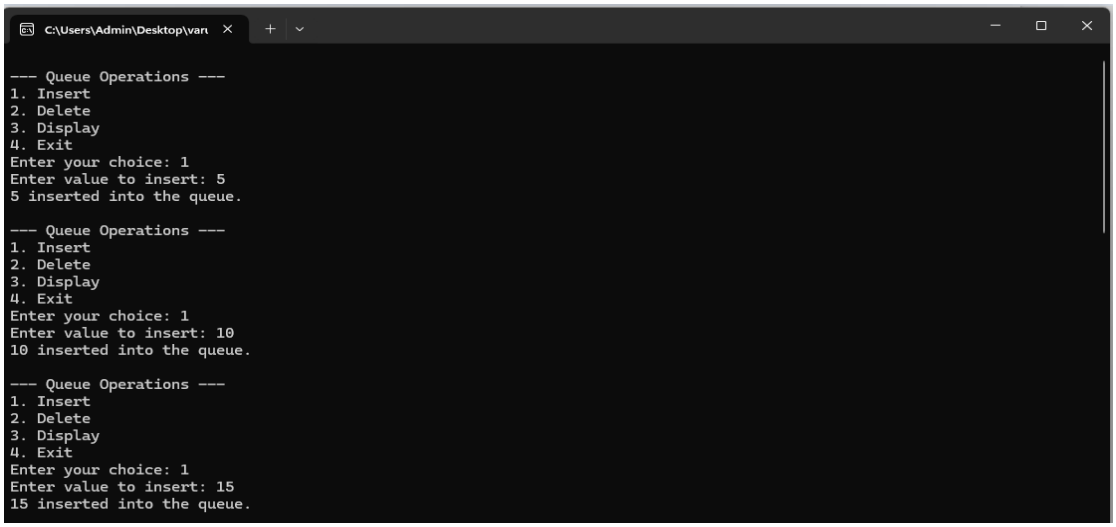
LAB PROGRAM - 03

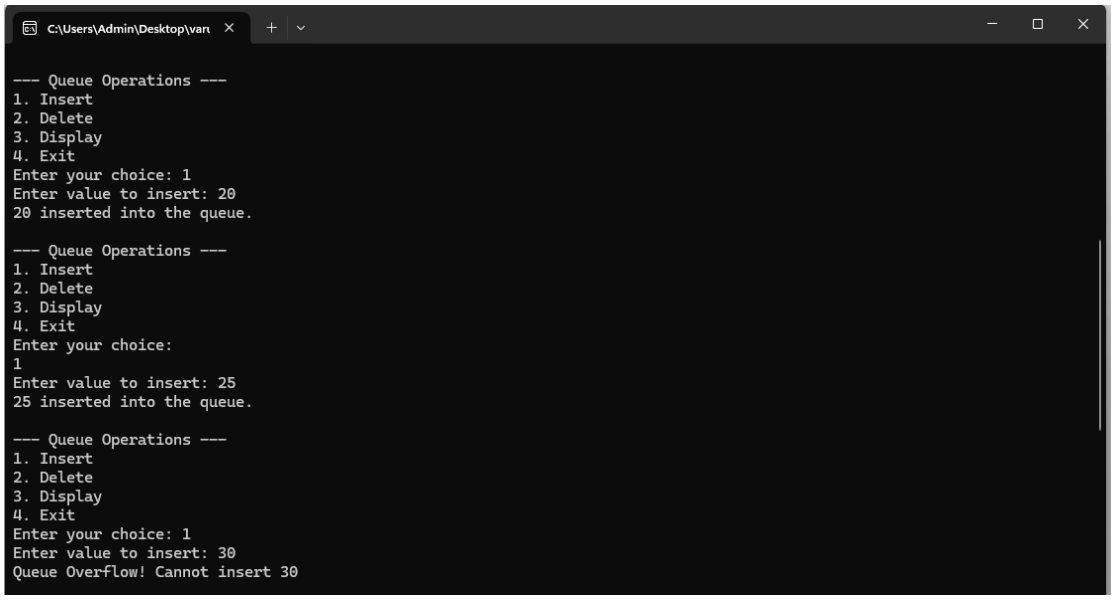
3. a) WAP to simulate the working of a queue of integers using an array. Provide the following operations: Insert, Delete, Display The program should print appropriate messages for queue empty and queue overflow conditions.

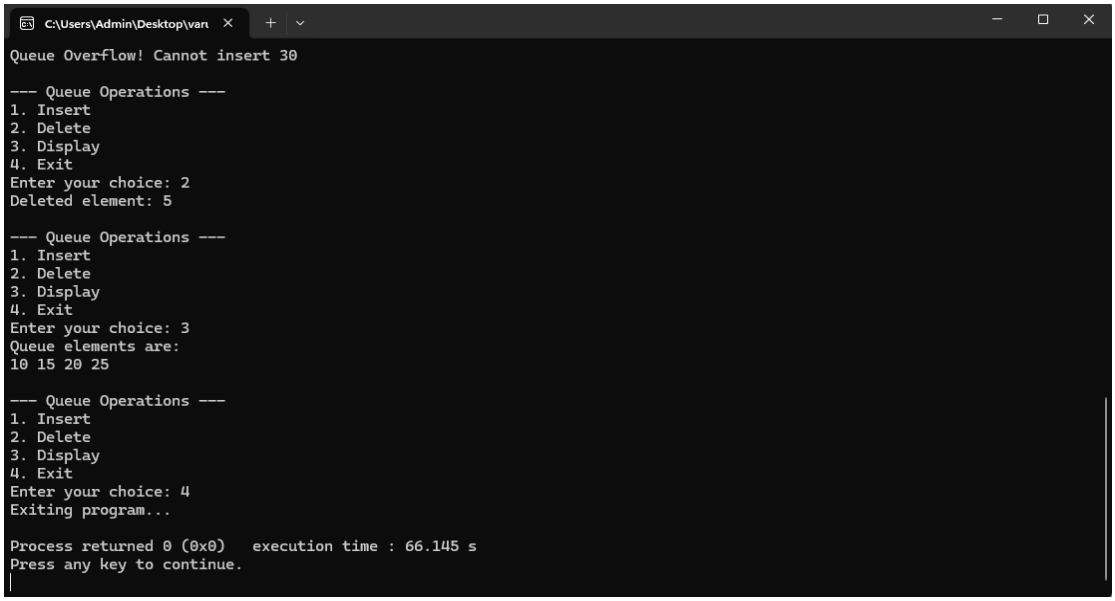
```
#include <stdio.h>
#define MAX 5   // Maximum size of the queue

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

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

// Function to delete an element (dequeue)
void delete() {
   if (front == -1 || front > rear) {
      printf("Queue Underflow! Queue is empty.\n");
   } else {
      printf("Deleted element: %d\n", queue[front]);
      front = front + 1;
   }
}

// Function to display all elements
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, value;

   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:
            printf("Enter value to insert: ");
            scanf("%d", &value);
            insert(value);
            break;

         case 2:
            delete();
            break;

         case 3:
            display();
            break;

         case 4:
            printf("Exiting program...\n");
            return 0;

         default:
            printf("Invalid choice! Try again.\n");
      }
   }
}
```

Output:


C:\Users\Admin\Desktop\var
--- Queue Operations ---
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 1
Enter value to insert: 5
5 inserted into the queue.
--- Queue Operations ---
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 1
Enter value to insert: 10
10 inserted into the queue.
--- Queue Operations ---
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 1
Enter value to insert: 15
15 inserted into the queue.


C:\Users\Admin\Desktop\var
--- Queue Operations ---
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 1
Enter value to insert: 20
20 inserted into the queue.
--- Queue Operations ---
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice:
1
Enter value to insert: 25
25 inserted into the queue.
--- Queue Operations ---
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 1
Enter value to insert: 30
Queue Overflow! Cannot insert 30


C:\Users\Admin\Desktop\var
Queue Overflow! Cannot insert 30
--- Queue Operations ---
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 2
Deleted element: 5
--- Queue Operations ---
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 3
Queue elements are:
10 15 20 25
--- Queue Operations ---
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 4
Exiting program...
Process returned 0 (0x0) execution time : 66.145 s
Press any key to continue.