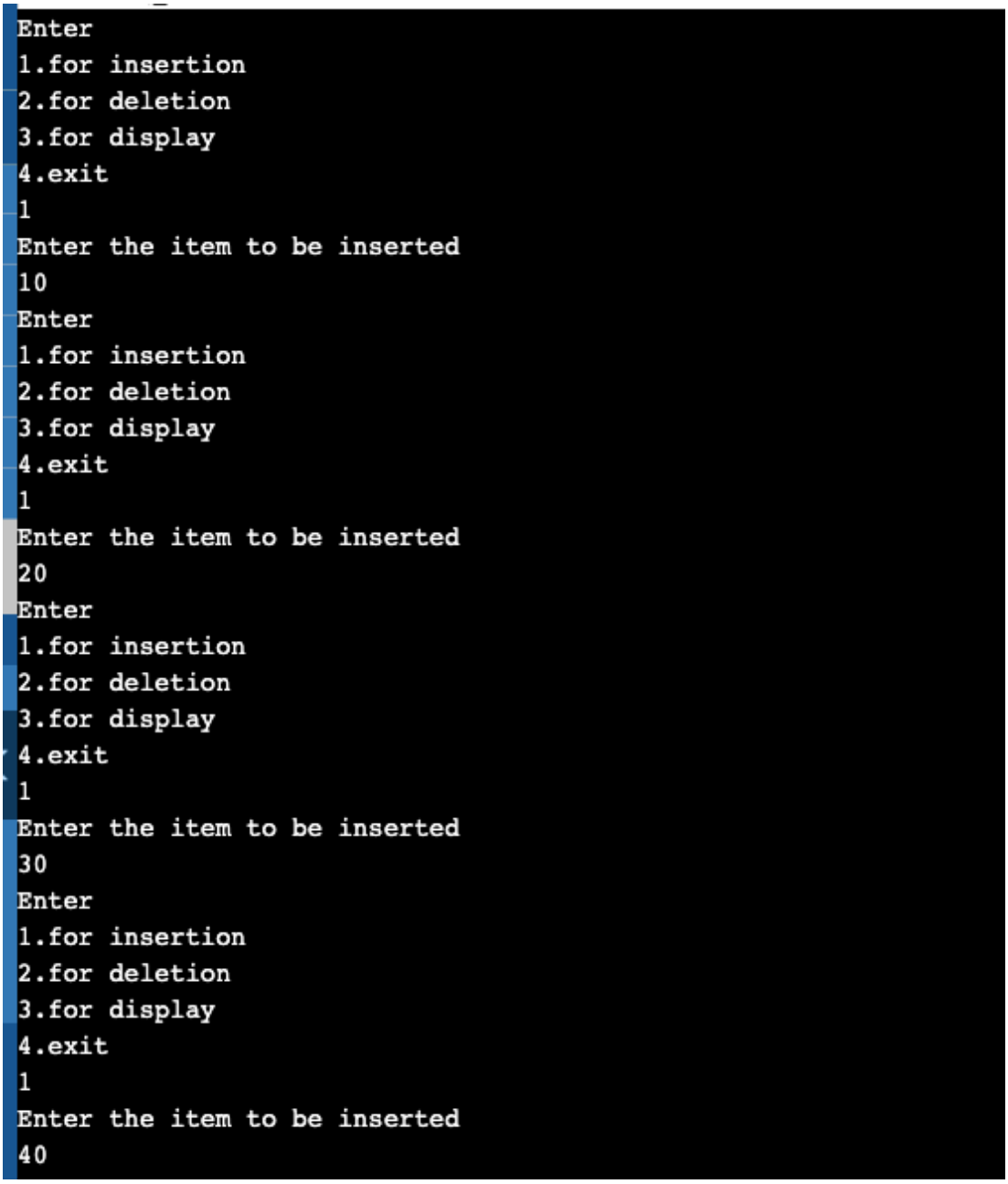
Program: Implementation Queues in C

```
#include<stdio.h>
#include<stdlib.h>
#define QUE_SIZE 5
int item,front=0,rear=-1,q[10];
void insertrear()
{
   if(rear == QUE_SIZE - 1)
   {
      printf("-------------------\n");
      printf("Queue OVERFLOW!!\n");
      printf("-------------------\n");
      return;
   }
   rear++;
   q[rear] = item;
}
int deletefront()
{
   if(front>rear)
   {
      front = 0;
      rear = -1;
      return -1;
   }
   return q[front++];
}
void displayQ()
{
   if(front>rear)
   {
      printf("-------------------\n");
      printf("Queue is empty\n");
      printf("-------------------\n");
      return;
   }
   printf("Contents of Queue\n");
   for(int i = front;i<=rear;i++)
   {
      printf("%d\n",q[i]);
   }
}
void main()
{
   int choice;
   for(;;)
   {
      printf("Enter \n1.for insertion\n2.for deletion\n3.for display\n4.exit\n");
      scanf("%d",&choice);
      switch(choice)
      {
         case 1: printf("Enter the item to be inserted\n");
               scanf("%d",&item);
               insertrear();
```

```
            break;
        case 2: item = deletefront();
            if(item == -1)
            {
               printf("-------------------\n");
               printf("Queue is empty\n");
               printf("-------------------\n");
            }
            else
               printf("Item deleted = %d\n",item);
            break;
        case 3: displayQ();
            break;
        default: exit(0);
      }
   }
}
```

OUTPUT:



```
Enter
1.for insertion
2.for deletion
3.for display
4.exit
1
Enter the item to be inserted
50
Enter
1.for insertion
2.for deletion
3.for display
4.exit
2
Item deleted = 10
Enter
1.for insertion
2.for deletion
3.for display
4.exit
3
Contents of Queue
20
30
40
50
Enter
1.for insertion
2.for deletion
3.for display
4.exit
4
```