1. Merging of sorted array

```
■ File Edit Search Run Compile Debug Project Options
                                                                                    Window Help
                                                                                             3=[1]=
                                            MERGING.CPP =
 #include<stdio.h>
 #include<conio.h>
∨oid main()
 int array1[50],array2[50],array3[100],m,n,i,j,k=0;
 clrscr();
 printf("\n Enter the size of the array Array 1:");
scanf("\n', &m);
printf("\n Enter sorted elements of array 1: \n");
 for(i=0;i<m;i++)
 scanf("xd",&array1[i]);
 printf("\n Enter the size of the array Array 2:");
scanf("\d",&n);
printf("\n Enter sorted elements of array 2: \n");
 for(i=0;i<n;i++)
  scanf ("xd",&array2[i]);
        — 1:1 ——(I
F1 Help F2 Save F3 Open Alt-F9 Compile F9 Make F10 Menu
```

```
≡ File Edit Search Run Compile Debug Project Options
                                                               Window Help
                                                                      -3=[↑]<del>-</del>j
                                 MERGING.CPP =
 [ • ]=
 scanf("xd",&array2[i]);
 i=0;
 j=0;
 while(i<m&&j<n)
 if(array1[i]Karray2[i])
 array3[k]=array1[i];
 i++;
 else
 array3[k]=array2[j];
 j++;
 k++;
 if(i>=m)
       37:1 ——
F1 Help F2 Save F3 Open Alt-F9 Compile F9 Make F10 Menu
```

```
≡ File Edit Search Run Compile Debug Project Options Window Help
                                                                     MERGING.CPP =
  -[ • ]-
 if(i>=m)
 while(j∢n)
 array3[k]=array2[j];
  j++;
 k++;
 if(j>=n)
 while(i<m)
 array3[k]=array1[i];
 i++;
 k++;
 printf("\n After merging: \n");
 for(i=0;i<m+n;i++)
F1 Help F2 Save F3 Open Alt-F9 Compile F9 Make F10 Menu
```

```
File Edit Search Run Compile Debug Project Options Window Help

S.C.

MERGING.CPP

MERGING.CPP

3=[1]

MERGING.CPP

4=[1]

MER
```

```
Enter the size of the array Array 1:3

Enter sorted elements of array 1:
2 1 1

Enter the size of the array Array 2:3

Enter sorted elements of array 2:
2 1 1

After merging:
2
1
1
2
1
1
```

## 2. Circular queue

```
Window Help
    File Edit Search Run Compile Debug Project Options

CIRCULAR.C
#include<stdio.h>
#include<conio.h>
#define MAX5
int cqueue_arr[MAX];
int front = -1;
int rear = -1;
void insert(int item)
if((front == 0 && rear == MAX-1) || (front == rear+1))
printf("Queue overflow \n");
                                                                      return;
if (front == -1)
front = 0;
rear = 0;
else
if(rear == MAX-1)
F1 Help F2 Save F3 Open Alt-F9 Compile F9 Make F10 Menu
```

```
File Edit Search Run Compile Debug Project Options

CIRCULAR.C
                                                                     Window Help
if (rear == MAX-1)
 rear = 0;
else
rear = rear+1;
cqueue_arr[rear] = item;
void deletion()
if(front == -1)
                                                                  П
printf("Queue underflow\n");
return;
printf("Element deleted from queue is : xd/n",cqueue_arr[front]);
if(front == rear)
front = -1;
rear=-1;
F1 Help F2 Save F3 Open Alt-F9 Compile F9 Make F10 Menu
```

```
-1-[‡]-
-[•]-
front = -1:
rear=-1;
else
if(front == MAX-1)
front = 0;
else
front = front+1;
                                          П
void display()
int front_pos = front,rear_pos = rear;
if (front == -1)
printf("Queue is empty\n");
return;
printf("Queue element\n");
if( front_pos <= rear_pos )</pre>
```

```
File Edit Search Run Compile Debug Project Options

| CIRCULAR.C |
                                                                  Window Help
printf C
          eue element\n"):
if ( front_pos <= rear_pos )
while(front_pos <= rear_pos)
printf("xd",cqueue_arr[front_pos]);
front_pos++;
else
while(front_pos <= MAX-1)
                                                               п
printf("%d",cqueue_arr[front_pos]);
front_pos++;
front_pos = 0;
while(front_pos <= rear_pos)
printf("xd",cqueue_arr[front_pos]);
front_pos++;
F1 Help F2 Save F3 Open Alt-F9 Compile F9 Make F10 Menu
```

```
■ File Edit Search Run Compile Debug Project Options Window Help
                                                                                                                                                                                                                        = CIRCULAR.C =
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              -1-[‡]-
  printf("\n");
 void main()
 int choice, item:
 clrscr();
do
printf("1.Insert\n");
printf("2.Delete\n");
printf("3.Display\n");
printf("4.Quit\n");
printf("Enter your cho
                                                                                                                                                                                                                                                                                                                                                                                                                               П
                                                                                                                               choice :");
   scanf ("xd", &choice);
 switch(choice)
 case 1:
printf("Input the element for insertion in queue : "); scanf("xd", &item);
 insert(item);
 break;
97:1 To Provide the Figure 1 of the Figure 1 o
```

```
File Edit Search Run Compile Debug Project Options Window Help
                                   = CIRCULAR.C =
 case 1:
printf("Input the element for insertion in queue : ");
scanf("%d", &item);
 insert(item);
break;
case 2 :
deletion();
break:
case 3:
                                                                  П
 display();
break;
 case 4:
 break;
default:
 printf("Wrong choice\n");
while(choice!=4);
 getch();
F1 Help F2 Save F3 Open Alt-F9 Compile F9 Make F10 Menu
```

```
1.Insert
2.Delete
3.Display
4.Quit
Enter your choice :1
Input the element for insertion in queue : 2
1.Insert
2.Delete
3.Display
4.Quit
Enter your choice :1
Input the element for insertion in queue : 3
1.Insert
2.Delete
3.Display
4.Quit
Enter your choice :3
Queue element
1.Insert
2.Delete
3.Display
4.Quit
Enter your choice :_
```

```
Input the element for insertion in queue : 3
1.Insert
2.Delete
3.Display
4.Quit
Enter your choice :3
Queue element
23
1.Insert
2.Delete
3.Display
4.Quit
Enter your choice :2
Element deleted from queue is : 2/n1.Insert
2.Delete
3.Display
4.Quit
Enter your choice :3
Queue element
3
1.Insert
2.Delete
3.Display
4.Quit
Enter your choice :3
Queue element
3
1.Insert
2.Delete
3.Display
4.Quit
Enter your choice :
```