

Stack:

DSARRAY.c

```
1  #include<stdio.h>
2  #include<conio.h>
3  int stack[100];
4  int top = -1;
5  int size;
6  void push(){
7      int value;
8      printf("Enter Value: ");
9      scanf("%d",&value);
10     if(top >= size - 1){
11         printf("Stack Is Full. Cannot Push.\n");
12         return;
13     }
14     stack[++top] = value;
15     printf("Pushed %d onto the stack.\n",value);
16 }
17 void pop(){
18     if(top < 0){
19         printf("Stack is Empty. Cannot Pop.\n");
20         return;
21     }
22     int popvalue = stack[top--];
23     printf("Poped %d from the stack.\n",popvalue);
24 }
25 void peek(){
26     if(top < 0){
27         printf("Stack Is Empty. Cannot Peek.\n");
28         return;
29     }
30     printf("Top Element of the stack : %d\n",stack[top]);
31 }
```

DSARRAY.c

```
32 void isEmpty(){
33     if(top < 0){
34         printf("Stack Is Empty.\n");
35     }else{
36         printf("Stack Is Not Empty.\n");
37     }
38 }
39 void Fdisplay(){
40     if(top < 0){
41         printf("Stack Is Empty.\n");
42         return;
43     }
44     printf("Stack Elements: \n");
45     for(int i = top; i >= 0; i--){
46         printf("%d ",stack[i]);
47     }
48     printf("\n");
49 }
```

```
50 void Bdisplay(){
51     if(top < 0){
52         printf("Stack Is Empty.\n");
53         return;
54     }
55     printf("Stack Element: \n");
56     for(int i = 0; i <= top; i++){
57         printf("%d ",stack[i]);
58     }
59     printf("\n");
60 }
```

```
61 int main() {
62     printf("Enter The Size Of The Stack : ");
63     scanf("%d",&size);
64
65     int choice;
66     do{
67         printf("Enter Choice For Operation : ");
68         scanf("%d",&choice);
69         switch(choice){
70             case 1:
71                 push();
72                 break;
73             case 2:
74                 pop();
75                 break;
76             case 3:
77                 peek();
78                 break;
79             case 4:
80                 isEmpty();
81                 break;
82             case 5:
83                 Fdisplay();
84                 break;
85             case 6:
86                 Bdisplay();
87                 break;
88             default:
89                 printf("Invalid Choice!!!");
90                 break;
91         }
92     }while(choice != 0);
93 }
```

Queue:

DSARRAY.c

```
1  #include<stdio.h>
2  #include<conio.h>
3  #include<stdbool.h>
4  int queue[100];
5  int front = -1;
6  int rear = -1;
7  void enqueue(){
8      int element;
9      printf("\nEnter Element For Queue : \n");
10     scanf("%d",&element);
11     if(rear == 100 - 1){
12         printf("\nQueue Is Full. No Enqueue Process.\n");
13     }
14     if(front == -1){
15         front = 0;
16     }
17     rear++;
18     queue[rear] = element;
19 }
20 void dequeue(){
21     if(front == -1 || front > rear){
22         printf("\nQueue Is Empty\n");
23     }
24     int element = queue[front];
25     front++;
26     printf("\nRemoved Element Is : %d\n",element);
27 }
```

```
28 void frontelee(){
29     if(front == -1){
30         printf("\nQueue Is Empty.\n");
31     }
32     else{
33         printf("\nFront Element : %d\n",queue[front]);
34     }
35 }
36 void rearelee(){
37     if(front == -1){
38         printf("\nQueue Is Empty.\n");
39     }
40     else{
41         printf("\nRear Element : %d\n",queue[rear]);
42     }
43 }
44 void display(){
45     if(front == -1){
46         printf("\nQueue Is Empty.\n");
47     }
48     for(int i = front; i <= rear; i++){
49         printf("q[%d] : %d\n",i,queue[i]);
50     }
51     printf("\n");
52 }
```

```
53 int main(){
54     int choice;
55     do{
56         printf("\nEnter Choice For Operation : ");
57         scanf("%d",&choice);
58         switch(choice){
59             case 1:
60                 enqueue();
61                 break;
62             case 2:
63                 dequeue();
64                 break;
65             case 3:
66                 frontele();
67                 break;
68             case 4:
69                 rearerele();
70                 break;
71             case 5:
72                 display();
73                 break;
74             default:
75                 printf("Invalid Options!!!");
76                 break;
77         }
78     }while(choice != 0);
79 }
```

Circular Queue:

DSARRAY.c

```
1  #include<stdio.h>
2  #include<conio.h>
3  #include<stdbool.h>
4  int cqueue[6];
5  int front = -1;
6  int rear = -1;
7  void enqueue(){
8      int element;
9      printf("\nEnter Element : \n");
10     scanf("%d",&element);
11     if((front == rear + 1) || (front == 0 && rear == 6 - 1)){
12         printf("\nQueue Is Full. No Enqueue Process.\n");
13     }
14     else{
15         if(front == -1) front = 0;
16         rear = (rear + 1) % 6;
17         cqueue[rear] = element;
18         printf("\nInserted Element : %d\n",element);
19     }
20 }
21 void dequeue(){
22     int element;
23     if((front == -1) && (rear == -1)){
24         printf("\nQueue Is Empty.\n");
25     }
26     element = cqueue[front];
27     if(front == rear){
28         front = rear - 1;
29     }else{
30         front = (front + 1) % 6;
31     }
32     printf("\nDeleted Item Is : %d\n",element);
33 }
34 void isfull(){
35     if((front == rear + 1) || (front == 0 && rear == 6 - 1)){
36         printf("\nCircular Queue Is Full.\n");
37     }
38 }
39 void isempty(){
40     if(front == -1){
41         printf("\nCircular Queue Is Empty.\n");
42     }
43 }
44 void frontelet(){
45     printf("\nCircular Queue Front Element : %d\n",cqueue[front]);
46 }
47 void rearelet(){
48     printf("\nCircular Queue Rear Element : %d\n",cqueue[rear]);
49 }
```

```

50 void display(){
51     if(front == -1){
52         printf("\nCircular Queue Is Empty.\n");
53     }int i = front;
54     printf("\nCircular Queue Is : \n");
55     while(i != rear){
56         printf("%d ",cqueue[i]);
57         i = (i + 1) % 6;
58     }
59     printf("%d ",cqueue[rear]);
60 }
61 void nextele(){
62     if((front == -1) && (rear == -1)){
63         printf("\nQueue Is Empty.\n");
64     }else{
65         printf("\nCurrent Element : %d\n",cqueue[rear]);
66         int next = (front + 1) % 6;
67         printf("Next Element Is : %d\n",cqueue[next]);
68     }
69 }

```

```

1 int main() {
2     int choice;
3     do {
4         printf("\nEnter Choice For Operation : ");
5         scanf("%d",&choice);
6         switch(choice) {
7             case 1:
8                 enqueue();
9                 break;
10            case 2:
11                dequeue();
12                break;
13            case 3:
14                isfull();
15                break;
16            case 4:
17                isempty();
18                break;
19            case 5:
20                frontelet();
21                break;
22            case 6:
23                rearlet();
24                break;
25            case 7:
26                display();
27                break;
28            case 8:
29                nextele();
30                break;
31            default:
32                printf("\nInvalid Option!!!\n");
33                break;
34        }
35    } while(choice != 0);

```