Double Queue:

```c
#include<stdio.h>
#define qsize 5
int f=0, r=-1, ch;
int item, q[10];

int isfull()
{
    return(r==qsize-1)?:0;
}

int isempty()
{
    return(f>r)?1:0;
}

void insert_rear()
{
    if(isfull())
    {
        printf("Queue of overflow \n");
        return;
    }
    r=r+1;
    q[r]=item;
}

void delete_front()
{
    if(isempty())
```

```
        {
            printf("Queue empty \n");
            return;
        }
        printf(" Item deleted is %d \n", q[f++]);
        if (f > r)
        {
            f = 0;
            r = 1;
        }
    }


void insert_front
    {
        if (f != 0)
        {
            f = f - 1;
            q[f] = item;
            return;
        }
        else if ((f == 0) && (r == -1))
        {
            q[++(r)] = item;
            return;
        }
        else
            printf("Insertion not possible \n");
    }


void delete_rear()
    {
        if (isempty())
        {
```

```c
        printf("Item deleted is %d \n", q[(r)--]);
        if (f > r)
        {
            f = 0;
            r = -1;
        }
    }
}

void display()
{
    int i;
    if (isempty())
    {
        printf("Queue is empty \n");
        return;
    }
    for (i = f; i <= r; i++)
    printf("%d \n", q[i]);
}

int main()
{
    for (;;)
    {
        printf("_____");
        printf(" 1) Insert rear \n2) Insert front
                \n 3) Delete rear \n 4) Delete
                front \n 5) Display \n 6) Exit \n");
        printf("Enter choice \n");
        scanf("%d", &ch);
        switch (ch)
        {
            case 1: printf("Enter the item \n");
                    scanf("%d", &item);
```

```
                insert rear ();
                break;
     case 2 : printf ("Enter the item \n");
             scanf ("%d";& item);
             insert front ();
             break;
     case 3 : delete_rear ();
             break;
     case 4 : delete_front ();
             break;
     case 5 : display ();
             break;
     default : exit (0);
        }
     }

     return 0;
}
```