```
#include<stdio.h>
#include<stdlib.h>
struct node
{
    int data;
    struct node *next;
};
struct node *front=NULL,*rear=NULL;
void enqueue(void);
void dequeue(void);
void peek(void);
void display(void);
void main()
{
    int ch;
    while(1)
    {
        printf("1.Enqueue\n");
        printf("2.Dequeue\n");
        printf("3.Peek\n");
        printf("4.Display\n");
        printf("5.Exit\n");
        printf("Enter your choice ");
        scanf("%d",&ch);
        switch(ch)
        {
            case 1: enqueue();
                    break;
            case 2: dequeue();
                    break;
            case 3: peek();
                    break;
            case 4: display();
                    break;
            case 5: exit(0);
            default: printf("Invalid Input\n");
        }
    }
}
void enqueue()
{
    int info;
    struct node *newnode;
    newnode=(struct node *)malloc(sizeof(struct node));
    if(newnode==NULL)
    {
        printf("Malloc error\n");
        return;
    }
    printf("Enter your data ");
    scanf("%d",&info);
    newnode->data=info;
    newnode->next=NULL;
    if(front==NULL&&rear==NULL)
    {
        front=rear=newnode;
    }
    else
```

```
    {
        rear->next=newnode;
        rear=newnode;
    }
}
void dequeue()
{
    struct node *temp=front;
    if(front==NULL&&rear==NULL)
    {
        printf("Queue is empty\n");
        return;
    }
    front=front->next;
    free(temp);
}
void peek()
{
    if(front==NULL&&rear==NULL)
    {
        printf("Queue is empty\n");
        return;
    }
    printf("%d\n",front->data);
}
void display()
{
    struct node *temp=front;
    if(front==NULL&&rear==NULL)
    {
        printf("Queue is empty\n");
        return;
    }
    while(temp!=NULL)
    {
        printf("%d ",temp->data);
        temp=temp->next;
    }
    printf("\n");
}
```