Linear queue code

```
#include <stdio.h>
#define N 5;
int front=-1;
int rear = -1;
int queue[N];
void enqueue(int x)
{     if (rear==N-1)
           printf("queue overflow\n");
      else if (front == -1 && rear== -1)
      {
           front=rear=0;
           queue[rear]=x;
      }
      else
      {     rear++;
            queue[rear]=x;
      }
}
void dequeue()
{     if (front==-1 && rear==-1)
           printf("queue is empty\n");
      else if (front==rear)
           front=rear=-1;
      else
           printf("removed value :%d\n");
}.
void display()
{   if (front==-1 && rear==-1)
         printf("queue is empty\n");
    else
    {  for (int i=front ; i<=rear ; i++)
              printf("%d\n", queue[i]);
    }
}
```

```
void main( )
{
  int ch,a;
  do{
     printf("enter choice : 1.enqueue 2.dequeue 3.display\n");
     scanf("%d ", &ch);
     switch(ch)
     {
        case1 : {printf("enter the number:\n");
                 scanf("%d",&a);
                 enqueue(a);
                 break;}
        Case 2: { dequeue( );
                 break;}
        Case 3: { display( );
                 break;}
        default: printf("invalid choice\n");
     }
  }while (ch!=0);
}
```

Output:

```
enter choice: 1.enqueue 2.dequeue 3.display
1
enter the number
10
enter choice : 1.enqueue 2.dequeue 3.display
1
enter the number
20
enter choice : 1.enquee 2.dequeue 3.display
1
enter the number
30
enter choice : 1.enqueue 2.dequeue 3.display
1
enter the number
40
enter choice : 1.enqueue 2.dequeue 3.display
1
enter the number
50
enter choice : 1.enqueue 2.dequeue 3.display
2
removed value:10
enter choice : 1.enqueue 2.dequeue 3.display
3
20
30
40
50
enter choice: 1.enqueue 2.dequeue 3.display
0
invalid choice.
```

Circular queue code

```
#include <stdio.h>
#define N 5
int front=-1;
int rear=-1;
int queue[N];
void enqueue (int x)
{ if(front==-1 && rear==-1)
   {
     rear =front=0;
     queue[rear] =x;
   }
   else if ((rear+1)%N==front)
     printf("queue overflow\n");
   else
   { rear =(rear+1)%N;
     queue[rear]=x;
   }
}
void dequeue( )
{  if (front==-1 && rear==-1)
       printf("removed value:%d\n", queue[front]);
   else if (front==rear)
       front =rear=-1;
   else
   { printf("removed value :%d\n", queue[front]);
     front =(front+1)%N;
   }
}
void display( )
{ if ( front==-1 && rear==-1)
      printf("queue is empty \n");
  else if (front <=rear){
      for (int i=front ; i<=rear; i++)
          printf("%d\n", queue[i]);
  }
  else {
      for(int i= front ; i<N; i++)
          printf("%d\n", queue[i]);
      for (int j=0 ; j<=rear ; j++)
          printf ("%d\n", queue[i]);
  }
}
```

```
void main()
{ int ch,a;
  do {
      printf("enter choice: 1.enqueue 2.dequeue 3.display \n");
      scanf("%d", &ch);
      switch(ch)
      {
        case 1 : { printf("enter the number:\n");
                   scanf("%d", a);
                   enqueue(a);
                   break; }
        case 2 : { dequeue();
                   break; }
        case 3 : { display();
                   break; }.
        default: printf("invalid choice");
      }
  } while (ch!=0);
}
```

output:

```
enter choice: 1.enqueue 2.dequeue 3.display
1
enter the number:
10
enter choice: 1.enqueue 2.dequeue 3.display
1
enter the number:
20
enter choice: 1.enqueue 2.dequeue 3.display
1
enter the number:
30
enter choice: 1.enqueue 2.dequeue 3.display
1
enter the number:
40.
enter choice: 1.enqueue 2.dequeue 3.display
1
enter the number:
50
enter choice: 1.enqueue 2.dequeue 3.display
1
enter the number:
60
queue overflow
enter ~~the en~~choice: 1.enqueue 2.dequeue 3.display
2
removed value : 10
enter choice: 1.enqueue 2.dequeue 3.display
2
removed value : 20
enter choice: 1.enqueue 2.dequeue 3.display
1
enter the number:
60
enter choice : 1.enqueue 2.dequeue 3.display
2
removed value : 30
enter choice : 1.enqueue 2.dequeue 3.display
1
enter the number:
70
enter choice : 1.enqueue 2.dequeue 3.display
3
40
50
60
70
enter choice : 1.enqueue 2.dequeue 3.display
0
invalid choice :
```

4/11/25