LAB 4

Dqueue

```
#include <stdio.h>
#define qsize 3
int f=0, r=-1, ch;
int item, q[10];
int isfull()
{
  return (r==qsize-1) ? 1:0;
}
int isempty()
{
  return (f>r) ? 1:0;
}
Void insert_rear()
{
  if (isfull())
  {
    printf("Queue 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) && (f == -1))
  {
     q[++(r)] = item;
     return;
  }
  else
  printf("Insertion not possible \n");
}
Void delete_rear()
{
   if (isempty())
   {
      printf("Queue is empty \n");
      return;
   }
printf("item deleted is %d \n", q[(r)--]);
   if (f > r)
   {
      f = 0;
      r = -1;
   }
}
```

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

Void main()
{
  int n = 1;
while (n!=0)
{
printf ("\n 1. insert rear"
        "\n 2. insert_front"
        "\n 3. delete_rear"
        "\n 4. delete_front"
        "\n 5. display"
        "\n 6. Exit");

  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);
}
}
}
```