Double ended queue

```
#include <stdio.h>
#include <conio.h>
#include <stdlib.h>
#include <process.h>
#define qsize 5
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 ("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 (is_empty())
{
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 is empty \n");
return;
}
for (i=f; i<=r; i++)
```

```
printf ("%d \n", q[i]);
}
void main ()
{
  for (;;)
  {
  printf (" 1. insert_rear \n 2: insert_front \n 3: delete
        rear \n 4. delete_front \n 5. display \n 6. exit \n")
printf ("enter choice : ");
 scanf ("%d", &ch);
 switch (ch)
 {
   case 1: printf ("enter the item : ");
    scanf ("%d", &item);
    insert_rear ();
    break;
case 2 : printf ("enter the item \n");
    scanf ("%d", &item);
     insert_front ();
      break;
case 3: display ();
      break;
default : exit(0);
 }
  }
}
```