Date __/__/__
Page ____
SPLASH

Double queue

```
# include <stdio.h>
# include <stdlib.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("que overflow\n");
         return;
        }
 r  = r+1;
 q[r] = item;
 }
void delete_front()
 {
   if (isempty())
           {
              printf("queue empty\n").
              return;
```

Date __/__/__
Page ____
SPLASH

```
    }
    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;
        }
            q[++(r)] = item;
            return;
        }
        else
        printf ("insertion not
                          possible \n");
    }
void delete_rear()
  {
    if (isempty())
        {
        printf ("queue is empty \n");
        return;
        }
  printf ("item delete 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 \n", q[i]);
  }
int main ()
  {
    for(;;)
{
printf ("1. insert_rear\n 2. insert_front
        \n 3.delete_front \n 5.display
        \n 6. exit \n");
printf ("Enter choice\n");
Scanf ("%d", &ch);
Switch (ch)
{
  case1: printf ("Enter the item \n");
         Scanf ("%d", & item);
         insert_rear ();
         break;
case 2: printf ("enter the item\n");
        scanf ("%d", & item);
        insert _ front ();
         break;
case3: delete_ rear ();
           break;
case 4: delete- front ();
           break;
case 5: display ();
           break;
default: exit (0);
 }
}
return 0;
}
```

Date / /
Page
SPLASH

Multi queue

```
# include <stdio.h>
# include <stdlib.h>
# define N 3
int queue [3][N];
int front[3] = {0,0,0};
int rear [3] = {-1,-1,-1};
int item, pr;
void pq insert (int);
void pq delete ();
void display ();
int main ()
 {
   int ch;
   for (;;)
   {
     printf ("\n PRIORITY QUEUE\n");
     printf ("     **********    ");
     printf ("\n\t 1: Pq-insert\n");
     printf ("\n\t 2: Pq- delete\n");
      printf ("\n\t 3: Pq- display \n");
      printf ("\n\t4 Exit \n");
     printf (" Enter Choice :\n");
     Scanf ("%d", &ch);
      Switch (ch)
     {
       case 1: printf ("\n Enter priority
                          number \n"
                 Scanf ("%d", &pr);
               if (pr>0 && pr <4)
               pq insert (pr-1);
else
printf("only three priority exits
        1 2 3\n");
  break;
 case 2:
  pq delete();
   break;
  case 3:
  display ();
    break;
  case 4: exit(0);
 }
 }
 return 0;
}
void pqinsert (int pr)
 {
   if (rear [pr] = = N-1)
   printf ("\n Queue overflow");
    else
   {
    printf ("\n Enter the item\n");
    Scanf ("%d", &item);
    rear [pr]++;
    queue [pr] [rear [pr]] = item;
  }
}
void pqdelete ()
 {
   int i;
   for (i=0; i<3; i++)
     {
```

Date
Page
SPLASH

```
   if (rear[i] == front[i] - 1)
   printf("\n Queue empty \n");
   else
   {
     printf("\n Deleted item is %d
             of queue[i][front
                      [i]], i+1);
     front[i]++;
   }
 }
}

void display()
{
   int i, j;
   for (i=0; i<3; i++)
   {
    if (rear[i] == front[i] - 1)
    printf("\n Queue empty %d\n", i+1);
    else
    {
     printf("\n Queue %d", i+1);
     for (j = front[i]; j <= rear[i]; j++)
     printf("%d\t", queue[i][j]);
   }
 }
}
```