Lab-7 Practice Program.

*) Implement Priority Queue :-

```
#include <stdio.h>
#define N 3
int queue[3][N];
int front[3] = {0,0,0};
int rear[3] = {-1,-1,-1};
int item, pr;
void main()
{
  int ch;
while(1)
{
printf("Priority queue\n");
printf("**********\n");
printf("\n\t1: PQ insert\n");
printf("\n\t2: PQ delete\n");
printf("\n\t3: PQ display\n");
printf("\n\t4: Exit\n");
printf("\n Enter the choice\n");
```

```
        scanf("%d", &ch);
        switch (ch)
        {
case 1: printf("\n Enter the priority number\n");
        scanf("%d", &pr);
        if (pr > 0 && pr < 4)
        pqinsert (pr-1);
        else
        printf("\n Only 3 priority exists 1 2 3 \n");
        break;
case 2: pqdelete();
        break;
case 3: display();
        break;
case 4: exit(0);
        }
}
pqinsert (int pr)
{
  if (rear[pr] == N-1)
printf("\n Queue Overflow\n");
  else
{
printf("Enter the item");
scanf("%d", &item);
```

```
        rear[pr]++;
        queue[pr][rear[pr]] = item;
      }
    return;
  }
  pqdelete()
    {
      int i;
      for(i=0; i<3; i++)
      {
        if(rear[i] == front[i]-1)
      printf("\n Queue empty \n");
      else
      {
    printf("Deleted item is %d of queue %d\n",
        queue[i][front[i]], i+1);
    front[i]++;
     return;
    }
  }
}
 display()
 {
   int i, j;
   for(i=0; i<3; i++)
```

```
{
  if (rear[i] == front[i]-1)
  printf ("\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]);
  }
}
return;
}
```