Descending Priority Queue:-

```
#include<stdio.h>
#include<stdlib.h>
#define SIZE 3

int PQ[SIZE];
int count=0;
int flag=0;
void insert_rear(int Pr)
{
  int i=0;
  if(count==SIZE)
  {
    Printf("Queue overflow\n");
    return;
  }
  if(count==0)
  {
    PQ[count++]=Pr;
  }
else
  {
  for(i=count-1; i>=0; i--)
  {
    if(Pr>PQ[i])
    {
      PQ[i+1]=PQ[i];
    }
    else
    { break;
```

```
    PQ[i+1]=PR;
    count++;
}
int Remove_Big()
{
  return PQ[flag++];
}
void display()
{
  int i;
  if(count==0)
  {
    Printf("Queue is empty\n");
    return;
  }
  Printf("Contents of the Queue:\n");
  for(i=flag; i<count; i++)
  {
    Printf("%d\n", PQ[i]);
  }
}
int main()
{
  int choice, item;
  for(;;)
  {
    Printf("\n 1: Insert_Rear \n 2: Remove_Big
           \n 3: Display \n 4: Exit\n");
```

```
Printf("Enter the choice :\n");
Scanf("%d",&choice);
switch(choice)
{
 case 1:Printf("Enter the item to be inserted :\n");
        scanf("%d", &item);
        insert_rear(item);
        break;
 case 2 : item = Remove_Big();
        if(item==-1)
        Printf("Queue is Empty\n");
        else
        Printf("Item Deleted = %d \n", item);
        break;
 case 3 : display();
        break;
 default : exit(0);
 }
}
}
```