```
main.c
  1 /*ASCENDING PRIORITY QUEUE*/
  2 #include<stdio.h>
    #include<stdlib.h>
     #define QUE SIZE 5
     int item,rear=-1, q[QUE_SIZE],count=0;
     void insertrear()
  7--
  8
         if(rear== QUE SIZE-1)
  9 -
 10
              printf("Queue overflow \n");
 11
              return ;
 12
 13
 14
         rear=rear+1;
 15
         q[rear]= item;
 16
         count++;
 17
 18
 19
 20
     int deleteasc()
 21 - {
 22
         int small=99;
 23
         int spos=-1;
 24
         if(count==0)
 25 -
              {return -1;
 26
         for(int i=0;i<QUE_SIZE;i++)
 27
 28 -
 29
              if(q[i]<small)</pre>
 30 -
                small=q[i];
 31
 32
                spos=i;
 33
 34
 35
          }
 36
 37
         q[spos]=99;
 38
         count=count-1;
 39
         return small;
```

```
nain.c
 32
 33
 34
 35
         q[spos]=99;
         count=count-1;
 37
         return small;
 38
 40
     void display()
 41
 42 - {
         int i;
 43
         if(count==0)
 44
 45 =
              printf("Queue empty \n");
 46
              return;
 47
 49
 50
         printf("Contents of queue \n");
 51
         for(i=0; i<QUE_SIZE; i++)
 52
             if(q[i]==99)
 54
             continue;
             else
             printf("%d ",q[i]);
 57
 58
     }
 60
     void main()
 61
 62 * {
         int choice;
         for(;;)
 64
 65 +
              printf("\nENTER 1.insert rear 2.delete 3.display \n");
              scanf("%d", &choice);
 67
              switch(choice)
                  case 1: printf("Enter the item\n");
 70
                             anf("%d". &item);
```

```
main.c
  47
               return;
  48
  49
  50
  51
           printf("Contents of queue \n");
           for(i=0; i<OUE SIZE; i++)
  52
  53 *
              if(q[i]=99)
  54
  55
               continue:
              else
               printf("%d ",q[i]);
  57
  58
  60
       void main()
  61
  62 -
           int choice;
  63
           for(;;)
  64
  65 *
               printf("\nENTER 1.insert rear 2.delete 3.display \n");
               scanf("%d", &choice);
  67
               switch(choice)
  68
                    case 1: printf("Enter the item\n");
    scanf("%d", &item);
  70
  71
                            insertrear();
  72
                            break;
  73
                    case 2: item= deleteasc();
  74
                            if(item == -1)
  75
                                 printf("Queue empty \n");
  76
                             else
  77
                                  printf("Item deleted is %d", item);
  78
                            break;
  79
                    case 3: display();
  80
                            break;
  81
                    default: exit(0);
  82
  83
  84
  85
  86
```

```
ENTER 1.insert rear 2.delete 3.display
Enter the item
11
ENTER 1.insert rear 2.delete 3.display
Enter the item
30
ENTER 1.insert rear 2.delete 3.display
Enter the item
55
ENTER 1.insert rear 2.delete 3.display
Enter the item
10
ENTER 1.insert rear 2.delete 3.display
Enter the item
90
ENTER 1.insert rear 2.delete 3.display
Contents of queue
11 30 55 10 90
ENTER 1.insert rear 2.delete 3.display
```

```
Enter the item
90
ENTER 1.insert rear 2.delete 3.display
Contents of queue
11 30 55 10 90
ENTER 1.insert rear 2.delete 3.display
Item deleted is 10
ENTER 1.insert rear 2.delete 3.display
Item deleted is 11
ENTER 1.insert rear 2.delete 3.display
Item deleted is 30
ENTER 1.insert rear 2.delete 3.display
Item deleted is 55
ENTER 1.insert rear 2.delete 3.display
Item deleted is 90
ENTER 1.insert rear 2.delete 3.display
Queue empty
ENTER 1.insert rear 2.delete 3.display
^C
... Program finished with exit code 130
Press ENTER to exit console.
```

```
main.c
   1 /*DESCENDING PRIORITY*/
     #include<stdio.h>
     #include<stdlib.h>
  4 #define QUE_SIZE 3
      int item,rear=-1, q[QUE_SIZE],count=0;
   5
   6
      void insertrear()
   7 - {
          if(rear== QUE_SIZE-1)
  8
  9 -
               printf("Queue overflow \n");
  10
  11
               return ;
  12
          }
  13
  14
          rear=rear+1;
  15
          q[rear]= item;
  16
          count++;
  17
  18
      }
  19
  20
     int deletedesc()
  21 - {
  22
          int largest=0;
  23
          int spos=-1;
  24
          if(count==0)
  25 -
              {return -1;
  26
          for(int i=0;i<QUE_SIZE;i++)
  27
  28 -
  29
               if(q[i]>largest)
  30 -
  31
                 largest=q[i];
  32
                 spos=i;
  33
  34
  35
          }
  36
  37
          q[spos]=0;
```

38

39

count=count-1;

return largest;

```
spos=1;
         q[spos]=0;
count=count-1;
          return largest;
     void display()
42 - {
         int i;
         if(count==0)
              printf("Queue empty \n");
              return;
         printf("Contents of queue \n");
         for(i=0; i<QUE_SIZE; i++)
            if(q[i]==0)
  continue;
             printf("%d ",q[i]);
    void main()
         int choice;
         for(;;)
             printf("\nENTER 1.insert rear 2.delete 3.display\n");
scanf("%d", &choice);
switch(choice)
                  case 1: printf("Enter the item \n");
```

```
4/
              return:
50
51
         printf("Contents of queue \n");
         for(i=0; i<QUE SIZE; i++)
52
53 -
54
            if(q[i]==0)
55
             continue;
56
            else
             printf("%d ",q[i]);
57
58
60
61
     void main()
62 -
63
         int choice;
         for(;;)
64
65 *
             printf("\nENTER 1.insert rear 2.delete 3.display\n");
66
                 nf("%d", &choice);
67
             switch(choice)
68
                  case 1: printf("Enter the item \n");
    scanf("%d", &item);
70
71
72
                          insertrear();
73
                          break;
74
                  case 2: item= deletedesc();
75
                          if(item == -1)
76
                               printf("Queue empty \n");
77
                          else
78
                               printf("Item deleted is %d", item);
79
                          break:
80
                 case 3: display();
81
                          break;
                 default: exit(0);
82
83
84
85
86
```

```
ENTER 1.insert rear 2.delete 3.display
Enter the item
30
ENTER 1.insert rear 2.delete 3.display
Enter the item
14
ENTER 1.insert rear 2.delete 3.display
Enter the item
80
ENTER 1.insert rear 2.delete 3.display
3
Contents of queue
30 14 80
ENTER 1.insert rear 2.delete
                              3.display
Item deleted is 80
ENTER 1.insert rear 2.delete
                              display
Item deleted is 30
ENTER 1.insert rear 2.delete
                              display
Item deleted is 14
ENTER 1.insert rear 2.delete
                              3.display
2
Queue empty
```

```
1 /*MULTIPLE PRIORITY QUEUE*/
 2 #include<stdio.h>
 3 #include<stdlib.h>
 4 #define N 3
   int queue[3][N];
   int front[3]={0,0,0};
 7 int rear[3]={-1,-1,-1};
 8 int item,pr;
   void main()
10 - {
11 int ch;
   while(1)
13 - {
printf("PRIORITY QUEUE:");
printf("\n1:PQinsert");
16 printf("\n2:PQdelete");
17 printf("\n3:PQdisplay");
   printf("\n4:Exit");
18
   printf("\nenter the choice\n");
19
20 scanf("%d", &ch);
21 switch(ch)
22 - {
     case 1:printf("enter the priority number\n");
23
24
            scanf("%d",&pr);
            if(pr>0 && pr<4)
25
            pqinsert(pr-1);
26
            else
27
28
                  ("only 3 priority exists 1 2 3\n");
29
            break:
    case 2:pqdelete();
31
           break:
32 case 3:display();
33
           break:
    case 4:exit(0);
35
36
37
    pqinsert(int pr)
```

```
37
    pqinsert(int pr)
39 - {
     if(rear[pr]==N-1)
40
41
     printf("Queue overflow\n");
42
     else
43 -
     printf("enter the item\n");
44
45
    scanf("%d",&item);
46
    rear[pr]++;
47
     queue[pr][rear[pr]]=item;
48
49
    return;
50
    pqdelete()
51
52 -
53 int i;
   for(i=0;i<3;i++)
55 {
      if(rear[i]==front[i]-1)
56
57
      printf("Queue empty\n");
58
      else
59 -
      printf("deleted item is %d of queue %d\n",queue[i][front[i]],i+1);
60
61
      front[i]++;
62
      return;
63
64
65
   display()
66
67 - 1
68 int i,j;
   for(i=0;i<3;i++)
70 -
71 if(rear[i]==front[i]-1)
     printf("Queue%d empty\n",i+1);
72
73
    else
74 -
      printf("\nOUEUE %d:".i+1):
```

```
printf("enter the item\n");
     scanf("%d",&item);
45
    rear[pr]++;
46
47
     queue[pr][rear[pr]]=item;
48
49
    return;
50
51
   pqdelete()
52 - {
53 int i;
54
   for(i=0;i<3;i++)
55 - {
56
      if(rear[i]==front[i]-1)
      printf("Queue empty\n");
57
58
      else
59 -
      printf("deleted item is %d of queue %d\n",queue[i][front[i]],i+1);
60
      front[i]++;
61
62
      return;
63
64
65
66 display()
67 - {
68 int i,j;
69 for(i=0;i<3;i++)
70 - {
71 if(rear[i]==front[i]-1)
72
           f("Queue%d empty\n",i+1);
73 else
74 -
      printf("\nQUEUE %d:",i+1);
75
      for(j=front[i];j<=rear[i];j++)</pre>
76
77
      printf("%d\t",queue[i][j]);
78
79
80
      return;
81
```

```
PRIORITY QUEUE:
1:PQinsert
2:PQdelete
3:PQdisplay
4:Exit
enter the choice
enter the priority number
enter the item
10
PRIORITY QUEUE:
1:PQinsert
2:PQdelete
3:PQdisplay
4:Exit
enter the choice
enter the priority number
enter the item
20
PRIORITY QUEUE:
1:PQinsert
2:PQdelete
3:PQdisplay
4:Exit
enter the choice
enter the priority number
enter the item
30
PRIORITY QUEUE:
1:PQinsert
2:PQdelete
3:PQdisplay
```

```
2:PQdelete
3:PQdisplay
4:Exit
enter the choice
deleted item is 10 of queue 1
PRIORITY QUEUE:
1:PQinsert
2:PQdelete
3:PQdisplay
4:Exit
enter the choice
enter the priority number
enter the item
50
PRIORITY QUEUE:
1:PQinsert
2:PQdelete
3:PQdisplay
4:Exit
enter the choice
enter the priority number
enter the item
PRIORITY QUEUE:
1:PQinsert
2:PQdelete
3:PQdisplay
4:Exit
enter the choice
enter the priority number
```

```
1
enter the priority number
enter the item
60
PRIORITY QUEUE:
1:PQinsert
2:PQdelete
3:PQdisplay
4:Exit
enter the choice
enter the priority number
enter the item
80
PRIORITY QUEUE:
1:PQinsert
2:PQdelete
3:PQdisplay
4:Exit
enter the choice
QUEUE 1:20
               30
QUEUE 2:50
                       80
                               Queue3 empty
               60
PRIORITY QUEUE:
1:PQinsert
2:PQdelete
3:PQdisplay
4:Exit
enter the choice
... Program finished with exit code 0
Press ENTER to exit console.
```