DS LAB-PROG-Extra Porgs

Multiple priority queue, ascending and descending priority

Program and Output

Mallika Prasad

1BM19CS081

4.11.2020

Program1-

Multiple Priority Queue

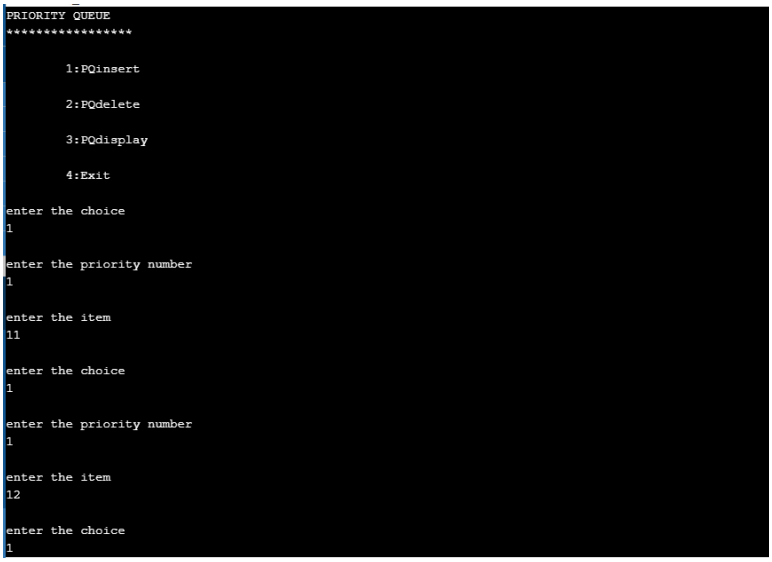
```
#include<stdio.h>
#define N 3
int queue[3][N];
int front[3]={0,0,0};
int rear[3]={-1,-1,-1};
int item,pr;

void pqinsert(int pr)
{
 if(rear[pr]==N-1)
 printf("\n Queue overflow\n");
 else
 {
 printf("\nenter the item\n");
 scanf("%d",&item);
 rear[pr]++;
 queue[pr][rear[pr]]=item;
```

```
 }
 return;
}

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

void 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("\nQUEUE %d:",i+1);
  for(j=front[i];j<=rear[i];j++)
    printf("%d\t",queue[i][j]);
  }
}
  return;
}
int main()
{
int ch;
printf("PRIORITY QUEUE\n");
printf("*****************\n");
printf("\n\t1:PQinsert\n");
printf("\n\t2:PQdelete\n");
printf("\n\t3:PQdisplay\n");
printf("\n\t4:Exit\n");
while(1)
{
printf("\nenter the choice\n");
scanf("%d",&ch);
switch(ch)
{
 case 1:printf("\nenter the priority number\n");
```

```
                scanf("%d",&pr);
                if(pr>0 && pr<4)
                pqinsert(pr-1);
                else
                printf("only 3 priority exists 1 2 3\n");
                break;
case 2:pqdelete();
        break;
case 3:display();
        break;
case 4:break;
}
}
return 0;
}
```

Output-



```
enter the priority number
1

enter the item
13

enter the choice
3

QUEUE 1:11      12      13      queue empty 2
queue empty 3

enter the choice
1

enter the priority number
1

 Queue overflow

enter the choice
1

enter the priority number
2

enter the item
21

enter the choice
1
```

```
enter the priority number
2

enter the item
22

enter the choice
1

enter the priority number
2

enter the item
23

enter the choice
3

QUEUE 1:11      12      13
QUEUE 2:21      22      23      queue empty 3

enter the choice
1

enter the priority number
3

enter the item
31

enter the choice
```

```
enter the choice
1

enter the priority number
3

enter the item
32

enter the choice
1

enter the priority number
3

enter the item
33

enter the choice
3

QUEUE 1:11      12      13
QUEUE 2:21      22      23
QUEUE 3:31      32      33
enter the choice
2
deleted item is 11 of queue 1

enter the choice
2
deleted item is 12 of queue 1
```

```
deleted item is 12 of queue 1

enter the choice
2
deleted item is 13 of queue 1

enter the choice
3
queue empty 1

QUEUE 2:21      22      23
QUEUE 3:31      32      33
enter the choice
1

enter the priority number
1

 Queue overflow

enter the choice
2
queue empty
deleted item is 21 of queue 2

enter the choice
2
queue empty
deleted item is 22 of queue 2
```

```
enter the choice
2
queue empty
deleted item is 23 of queue 2

enter the choice
3
queue empty 1
queue empty 2

QUEUE 3:31      32      33
enter the choice
1

enter the priority number
2

 Queue overflow

enter the choice
2
queue empty
queue empty
deleted item is 31 of queue 3

enter the choice
2
queue empty
queue empty
deleted item is 32 of queue 3
```

```
enter the choice
3
queue empty 1
queue empty 2

QUEUE 3:33
enter the choice
1

enter the priority number
3

 Queue overflow

enter the choice
2
queue empty
queue empty
deleted item is 33 of queue 3

enter the choice
3
queue empty 1
queue empty 2
queue empty 3

enter the choice
4


...Program finished with exit code 0
```

Program 2-

Ascending Priority

```
#include <stdio.h>
#define MAX 4
int pq[MAX];
int count = 0;
int d = 0;

void insert(int data){
  int i = 0;
    if(count==MAX)
    {
       printf("Queue overflow\n");
       return;
    }
    if(count == 0){
      pq[count++] = data;
    }else{
      for(i = count - 1; i >= 0; i-- ){
     if(data<pq[i]){
           pq[i+1] = pq[i];
         }else{
           break;
         }
       }
 pq[i+1] = data;
```

```
        count++;
      }
}

int removeData(){

  return pq[d++];
}
void display()
{int i;
if (count==0)
{
        printf("queue is empty\n");
        return;
}
printf("Contents of queue in ascending priority: ");
for(i=d;i<count;i++)
{
        printf("%d ",pq[i]);
}
printf("\n");
}

int main() {
   int choice,item;
        do
```

```
        {
                printf("\n1:insert 2:delete_smallest 3:display 4:exit\n");
                printf("Enter the choice :");
                scanf("%d",&choice);
                switch(choice)
                {
                        case 1:printf("Enter the item to be inserted :");
                        scanf("%d",&item);
                        insert(item);
                        break;
                        case 2:item=removeData();
                        if(item==-1)
                        printf("Queue is empty\n");
                        else
                        printf("item deleted=%d\n",item);
                        break;
                        case 3:display();
                        break;
                        default:break;
}

        }while(choice!=4);
return 0;
  }
```

Output-

```
1:insert 2:delete_smallest 3:display 4:exit
Enter the choice :3
queue is empty

1:insert 2:delete_smallest 3:display 4:exit
Enter the choice :1
Enter the item to be inserted :12

1:insert 2:delete_smallest 3:display 4:exit
Enter the choice :1
Enter the item to be inserted :14

1:insert 2:delete_smallest 3:display 4:exit
Enter the choice :1
Enter the item to be inserted :11

1:insert 2:delete_smallest 3:display 4:exit
Enter the choice :1
Enter the item to be inserted :15

1:insert 2:delete_smallest 3:display 4:exit
Enter the choice :3
Contents of queue in ascending priority: 11 12 14 15

1:insert 2:delete_smallest 3:display 4:exit
Enter the choice :2
item deleted=11
```

```
1:insert 2:delete_smallest 3:display 4:exit
Enter the choice :3
Contents of queue in ascending priority: 11 12 14 15

1:insert 2:delete_smallest 3:display 4:exit
Enter the choice :2
item deleted=11

1:insert 2:delete_smallest 3:display 4:exit
Enter the choice :2
item deleted=12

1:insert 2:delete_smallest 3:display 4:exit
Enter the choice :2
item deleted=14

1:insert 2:delete_smallest 3:display 4:exit
Enter the choice :2
item deleted=15

1:insert 2:delete_smallest 3:display 4:exit
Enter the choice :3
Contents of queue in ascending priority:

1:insert 2:delete_smallest 3:display 4:exit
Enter the choice :4


...Program finished with exit code 0
Press ENTER to exit console.
```

Program 3-

Descending Priority

```
#include <stdio.h>
#define MAX 4
int pq[MAX];
int count = 0;
int d = 0;

void insert(int data){
  int i = 0;
    if(count==MAX)
    {
       printf("Queue overflow\n");
       return;
    }
    if(count == 0){
      pq[count++] = data;
    }else{
      for(i = count - 1; i >= 0; i-- ){
       if(data>pq[i]){
           pq[i+1] = pq[i];
         }else{
           break;
         }
       }
```

```
            pq[i+1] = data;
        count++;
     }
}

int removeData(){
  return pq[d++];
}
void display()
{int i;
if (count==0)
{
        printf("queue is empty\n");
        return;
}
printf("Contents of queue in descending priority: ");
for(i=d;i<count;i++)
{
        printf("%d ",pq[i]);
}
printf("\n");
}

int main() {
   int choice,item;
        do
```

```
        {
                printf("\n1:insert 2:delete_largest 3:display 4:exit\n");
                printf("Enter the choice :");
                scanf("%d",&choice);
                switch(choice)
                {
                        case 1:printf("Enter the item to be inserted :");
                        scanf("%d",&item);
                        insert(item);
                        break;
                        case 2:item=removeData();
                        if(item==-1)
                        printf("Queue is empty\n");
                        else
                        printf("item deleted=%d\n",item);
                        break;
                        case 3:display();
                        break;
                        default:break;
}
}while(choice!=4);
return 0;
  }
```

Output-

```
1:insert 2:delete_largest 3:display 4:exit
Enter the choice :3
queue is empty

1:insert 2:delete_largest 3:display 4:exit
Enter the choice :1
Enter the item to be inserted :11

1:insert 2:delete_largest 3:display 4:exit
Enter the choice :1
Enter the item to be inserted :13

1:insert 2:delete_largest 3:display 4:exit
Enter the choice :1
Enter the item to be inserted :15

1:insert 2:delete_largest 3:display 4:exit
Enter the choice :1
Enter the item to be inserted :14

1:insert 2:delete_largest 3:display 4:exit
Enter the choice :3
Contents of queue in descending priority: 15 14 13 11

1:insert 2:delete_largest 3:display 4:exit
Enter the choice :2
item deleted=15

1:insert 2:delete_largest 3:display 4:exit
Enter the choice :2
```

```
1:insert 2:delete_largest 3:display 4:exit
Enter the choice :3
Contents of queue in descending priority: 15 14 13 11

1:insert 2:delete_largest 3:display 4:exit
Enter the choice :2
item deleted=15

1:insert 2:delete_largest 3:display 4:exit
Enter the choice :2
item deleted=14

1:insert 2:delete_largest 3:display 4:exit
Enter the choice :2
item deleted=13

1:insert 2:delete_largest 3:display 4:exit
Enter the choice :2
item deleted=11

1:insert 2:delete_largest 3:display 4:exit
Enter the choice :3
Contents of queue in descending priority:

1:insert 2:delete_largest 3:display 4:exit
Enter the choice :4


...Program finished with exit code 0
Press ENTER to exit console.
```