#include<stdio.h>

#define MAX 10

struct P\_Queue

{

int data;

int priority;

};

int main()

{

struct P\_Queue p[MAX];

int choice,prior,ele,i,rear=-1;

do

{

printf("Option 1.Insertion 2.Deletion 3.Display");

scanf("%d",&choice);

switch(choice)

{

case 1:

printf("Enter the data:");

scanf("%d",&ele);

printf("Enter the priority");

scanf("%d",&prior);

if(rear==-1){

p[++rear].data=ele;

p[rear].priority=prior;

}

else{

p[++rear].data=ele;

p[rear].priority=prior;

}

Sort(p,rear);

break;

case 2:

if(rear==-1)

printf("Queue is empty");

else

{

for(i=0;i<=rear;i++){

p[i].data = p[i+1].data;

p[i].priority=p[i+1].priority;

}

rear--;

}

break;

case 3:

if(rear==-1)

printf("Queue is empty");

else{

for(i=0;i<=rear;i++)

{

printf("%d %d",p[i].data,p[i].priority);

printf("\n");

}

}

break;

}

}while(choice<=3);

return 0;

}

void Sort(struct P\_Queue p[],int rear)

{

int i,j;

for(i=0;i<=rear;i++)

{

for(j=i+1;j<=rear;j++)

{

if(p[i].priority>p[j].priority){

int data=p[i].data;

int prior=p[i].priority;

p[i].data=p[j].data;

p[i].priority=p[j].priority;

p[j].data=data;

p[j].priority=prior;

}

}

}

}