#include<iostream>

using namespace std;

const int N=10;

class priorityQ

{

  int front, rear;

  int data\_element[N];

  int priority[N];

  public:

  priorityQ()

  {

      front=rear=-1;

  }

  void enqueue(int x,int pr)

  {

      if(rear==N-1)

      {

          cout<<"Queue is Full!"<<endl;

          return;

      }

      else if(front==-1 && rear==-1)

      {

          front=rear=0;

          data\_element[rear]=x;

          priority[rear]=pr;

      }

      else

      {

          int i;

          for(i=rear;i>=front;i--)

          {

              if(pr>=priority[i])

              {

                  data\_element[i+1]=data\_element[i];

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

              }

              else

              {

                  break;

              }

          }

          data\_element[i+1]=x;

          priority[i+1]=pr;

          rear++;

          cout<<"Element Enquequed"<<endl;

      }

  }

  void dequeue()

  {

      int temp,temp1;

      if(front==-1 && rear==-1)

      {

          cout<<"Queue is Empty!"<<endl;

          return;

      }

      else if(front==rear)

      {

          temp=data\_element[front];

          temp1=priority[front];

          front=rear=-1;

      }

      else

      {

          temp=data\_element[front];

          temp1=priority[front];

          front++;

      }

      cout<<"\n Element:"<<temp<<" with priority :"<<temp1<<" dequequed"<<endl;

  }

  void display()

  {

      if(front==-1 && rear==-1)

      {

          cout<<"Queue is empty!"<<endl;

      }

      else

      {

          cout<<"(Element Priority):"<<endl;

          for(int i=front;i<=rear;i++)

          {

              cout<<"("<<data\_element[i]<<","<<priority[i]<<")"<<endl;

          }

      }

  }

};

int main()

{

    priorityQ obj;

    int ch;

    int x,pr;

    do

    {

        cout<<"\nEnter option no. to perform operation on Priority Queue(Ascending):"<<endl;

        cout<<"Enter 0 for exit"<<endl;

        cout<<"1. Enqueue"<<endl;

        cout<<"2. Dequeue "<<endl;

        cout<<"3. Display"<<endl;

        cin>>ch;

        switch(ch)

        {

            case 0:

            break;

            case 1:

                cout<<"Enter Element:"<<endl;

                cin>>x;

                cout<<"Enter Priority:"<<endl;

                cin>>pr;

                obj.enqueue(x,pr);

                break;

            case 2:

                obj.dequeue();

                break;

            case 3:

                obj.display();

                break;

            default:

                cout<<"Enter the valid option!"<<endl;

                break;

        }

    }while(ch!=0);

    return 0;

}

Enter option no. to perform operation on Priority Queue(Ascending):

Enter 0 for exit

1. Enqueue

2. Dequeue

3. Display

1

Enter Element:

5

Enter Priority:

6

Enter option no. to perform operation on Priority Queue(Ascending):

Enter 0 for exit

1. Enqueue

2. Dequeue

3. Display

2

Element:5 with priority :6 dequequed

Enter option no. to perform operation on Priority Queue(Ascending):

Enter 0 for exit

1. Enqueue

2. Dequeue

3. Display

3

Queue is empty!

Enter option no. to perform operation on Priority Queue(Ascending):

Enter 0 for exit

1. Enqueue

2. Dequeue

3. Display

4

Enter the valid option!

Enter option no. to perform operation on Priority Queue(Ascending):

Enter 0 for exit

1. Enqueue

2. Dequeue

3. Display

0