#include<iostream>

using namespace std;

const int MAX=5;

class Job

{

int id;

int priority;

friend class Queue;

public:

void getdata()

{

cout<<"\nENter Job id: ";

cin>>id;

cout<<"\nENter Job priority: ";

cin>>priority;

}

void putdata()

{

cout<<"\n\t"<<id;

cout<<"\t\t"<<priority;

}

};

class Queue

{

int front,rear;

Job queue[MAX];

public:

Queue()

{

front=rear=-1;

}

bool isEmpty();

bool isFull();

void insert();

void remove();

void display();

};

bool Queue::isEmpty()

{

if(front==(rear+1)||rear==-1)

return 1;

else return 0;

}

bool Queue::isFull()

{

if(rear==MAX-1)

{

return 1;

}

else

return 0;

}

void Queue::insert()

{

Job j;

if(isFull())

{

cout<<"\nQueue is Full.";

}

else

{

j.getdata();

if(rear==-1)//empty

{

front++;

rear++;

queue[rear]=j;

}

else

{

int i=rear;

while(i>=front && queue[i].priority>j.priority)

{

queue[i+1]=queue[i];

i--;

}

queue[i+1]=j;

rear++;

}

cout<<"\nJob Added To Queue.";

}

}

void Queue::remove()

{

if(rear==-1||front==(rear+1))

{

cout<<"\nQueue is Empty.";

}

else

{

front++;

cout<<"\nJob Processed From Queue.";

}

}

void Queue::display()

{

if(isEmpty())

{

cout<<"\nQueue is Empty.";

}

else

{

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

{

queue[i].putdata();

}

}

}

int main()

{

int ch;

Queue q;

do

{

cout<<"\n\n\*\*\*\*MENU\*\*\*\*\n";

cout<<"1.Insert job\n";

cout<<"2.Display jobs\n";

cout<<"3.Remove job\n";

cout<<"4.Exit\n";

cout<<"Choice: ";

cin>>ch;

switch(ch)

{

case 1: q.insert();

break;

case 2: cout<<"\n\tJob id ";

cout<<"\t Job priority ";

q.display();

break;

case 3: q.remove();

}

}while(ch!=4);

return 0;

}

Output:

\*\*\*\*MENU\*\*\*\*

1.Insert job

2.Display jobs

3.Remove job

4.Exit

Choice: 1

ENter Job id: 2345

ENter Job priority: 2

Job Added To Queue.

\*\*\*\*MENU\*\*\*\*

1.Insert job

2.Display jobs

3.Remove job

4.Exit

Choice: 2

Job id Job priority

2345 2