```
//============================================================================
// Name        : 21465_Pract11.cpp
// Author      : Chaitanya Paraskar
// Roll No.    : 21465
// Aim         : Queues are frequently used in computer programming, and a
typical example is the creation
//               of a job queue by an operating system. If the operating system
does not use priorities, then
//               the jobs are processed in the order they enter the system.
Write C++ program for simulating job queue.
//               Write functions to add job, display job and delete job from
queue.
//============================================================================

#include <iostream>
using namespace std;

class p_queue
{
    int pid;
    int priority;
    p_queue *next;

public:
    p_queue(int p, int pr)
    {
        pid = p;
        priority = pr;
        next = NULL;
    }
    friend class Schedule;
};

class Schedule
{
    p_queue *front;
    p_queue *rear;

public:
    Schedule()
    {
        front = NULL;
        rear = NULL;
    }
    void push()
    {
        int p, q;
        cout << "enter priority of process";
        cin >> q;
        cout << "process id ";
        cin >> p;

        p_queue *n = new p_queue(p, q);
```

```
        if (front == NULL)
        {
            front = n;
            return;
        }
        else if (front->next == NULL)
        {
            if (n->priority < front->priority)
            {
                n->next = front;
                front = n;
                return;
            }
        }

        p_queue *temp = front;
        if (q < temp->priority)
        {
            n->next = front;
            front = n;
            return;
        }
        while (temp->next != NULL)
        {
            if (temp->next->priority > n->priority)
            {
                n->next = temp->next;
                temp->next = n;
                return;
            }
            temp = temp->next;
        }
        temp->next = n;
    }
    void traverse()
    {
        p_queue *temp = front;
        while (temp != NULL)
        {
            cout << temp->priority << " " << temp->pid << " " << endl;
            temp = temp->next;
        }
    }
    void del()
    {
        p_queue *temp = front;
        cout << "processs in exe " << temp->pid << " " << temp->priority <<
endl;
        front = front->next;
        delete temp;
    }
};
int main()
{
```

```
    Schedule obj;
    char c;
    do
    {
        int ch;
        cout << "1.Insert process\n2.Delete process\n enter your choice";
        cin >> ch;

        if (ch == 1)
        {
            obj.push();
            obj.traverse();
        }
        else if (ch == 2)
        {
            obj.del();
            obj.traverse();
        }
        else
            cout << "enter valid data<<endl";

        cout << "Do you want to continue(y)";
        cin >> c;

    } while (c == 'y');

    return 0;
}

/*

Output:

$ g++ Pract11.cpp -o out && ./out
1.Insert process
2.Delete process
enter your choice 1
enter priority of process 2
process id 111
2 111
Do you want to continue(y) y
1.Insert process
2.Delete process
enter your choice 1
enter priority of process 1
process id 222
1 222
2 111
Do you want to continue(y) y
1.Insert process
2.Delete process
enter your choice 1
enter priority of process 3
```

```
process id 333
1 222
2 111
3 333
Do you want to continue(y) y
1.Insert process
2.Delete process
enter your choice 2
processs in exe 222 1
2 111
3 333
Do you want to continue(y) y
1.Insert process
2.Delete process
enter your choice 2
processs in exe 111 2
3 333
Do you want to continue(y) y
1.Insert process
2.Delete process
enter your choice 2
processs in exe 333 3
Do you want to continue(y) y
1.Insert process
2.Delete process
enter your choice

*/
```