

```

//=====
// Name      : 21465_Pract12.cpp
// Author     : Chaitanya Paraskar
// Roll No.   : 21465
// Aim        : Write program to implement a priority queue in C++ using an
order list / array to store
//            the items in the queue.Create a class that includes the data
items(which should be template)
//            and the priority(which should be int).The order list / array
should contain these objects,
//            with operator<= overloaded so that the items with highest
priority appear at the beginning of
//            the list / array(which will make it relatively easy to retrieve
the highest item.)
//=====

#include <iostream>
using namespace std;

template <class T>
class p_queue
{
    T pid;
    int priority;
    p_queue *next;

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

    bool operator<(const p_queue<T> *other)
    {
        return this->priority < other->priority;
    }

    bool operator>(const p_queue<T> *other)
    {
        return this->priority > other->priority;
    }
};

class Schedule
{
    p_queue<int> *front;
    p_queue<int> *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<int> *n = new p_queue(p, q);

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

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

```

```

    }
}
void del()
{
    p_queue<int> *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++ Praact13.cpp -o out && ./out
```

Size of Queue = 10

Deque Menu:

1. Display Deque from Front to Back
2. Push Front
3. Push Back
4. Pop Front
5. Pop Back

6. Exit

Enter your choice: 1

Deque is empty

Deque Menu:

1. Display Deque from Front to Back

2. Push Front

3. Push Back

4. Pop Front

5. Pop Back

6. Exit

Enter your choice: 2

Enter data to push front: 123

123

Deque Menu:

1. Display Deque from Front to Back

2. Push Front

3. Push Back

4. Pop Front

5. Pop Back

6. Exit

Enter your choice: 2

Enter data to push front: 234

234 123

Deque Menu:

1. Display Deque from Front to Back

2. Push Front

3. Push Back

4. Pop Front

5. Pop Back

6. Exit

Enter your choice: 2

Enter data to push front: 3

3 234 123

Deque Menu:

1. Display Deque from Front to Back

2. Push Front

3. Push Back

4. Pop Front

5. Pop Back

6. Exit

Enter your choice: 3

Enter data to push back: 456

3 234 123 456

Deque Menu:

1. Display Deque from Front to Back

2. Push Front

3. Push Back

4. Pop Front

5. Pop Back

6. Exit

Enter your choice: 4

234 123 456

Deque Menu:

1. Display Deque from Front to Back

2. Push Front

3. Push Back

4. Pop Front

5. Pop Back

6. Exit

Enter your choice: 5

234 123

Deque Menu:

1. Display Deque from Front to Back

2. Push Front

3. Push Back

4. Pop Front

5. Pop Back

6. Exit

Enter your choice: 4

123

Deque Menu:

1. Display Deque from Front to Back

2. Push Front

3. Push Back

4. Pop Front

5. Pop Back

6. Exit

Enter your choice: 5

Deque is empty

Deque Menu:

1. Display Deque from Front to Back

2. Push Front

3. Push Back

4. Pop Front

5. Pop Back

6. Exit

Enter your choice: 6

Exiting Deque Menu

Deque Deleted

*/