

Roll No.: 64

Code:

```
#include<iostream>
#include <iostream>
#define MAX_SIZE 10

using namespace std;

class priority_queue {
private:
    string queue[MAX_SIZE];
    int priority_val[MAX_SIZE];
    int front;
    int rear;

public:
    priority_queue() {
        front = -1;
        rear = -1;
    }

    bool is_empty() { return front == -1; }

    bool is_full() { return rear == MAX_SIZE - 1; }

    void enqueue(string data, int priority) {
        if (is_full()) {
            cout << "Queue is full" << endl;
            return;
        }
        if (is_empty()) {
            front = 0;
            rear = 0;
            queue[rear] = data;
            priority_val[rear] = priority;
        } else {
            int i;
            rear++;
            for (i = rear-1; i >= front; i--) {
                if (priority_val[i] < priority) {
                    queue[i + 1] = queue[i];
```

```

        priority_val[i + 1] = priority_val[i];
    } else {
        break;
    }
}
queue[i + 1] = data;
priority_val[i + 1] = priority;
}
}
string dequeue() {
    if (is_empty()) {
        cout << "Queue is empty" << endl;
        return "";
    }
    string data = queue[front];
    if (front == rear) {
        front = -1;
        rear = -1;
    } else {
        front++;
    }
    return data;
}
int get_priority() {
    if (is_empty()) {
        return -1;
    }
    return priority_val[front];
}
void display() {
    if (is_empty()) {
        cout << "Queue is empty" << endl;
        return;
    }
    cout << "Queue is:" << endl;
    for (int i = front; i <= rear; i++) {
        cout << priority_val[i] << " : " << queue[i] <<
endl;
    }
    cout << endl;
}
};

```

```

int main() {
    priority_queue queue;
    int ch = 0;
    int priority;
    string name;
    cout<<"Priorities are 0: General Checkup, 1: Non-Serious,
2: Serious"<<endl;
    do {
        cout << "-----: MENU :-----" << endl;
        cout << "1. Add Patient" << endl;
        cout << "2. Remove Patient" << endl;
        cout << "3. Display Queue" << endl;
        cout << "4. Exit" << endl;
        cout << "Enter your choice: ";
        cin >> ch;
        switch (ch) {
            case 1:
                cout << "Enter Patient Name: ";
                cin >> name;
                cout << "Enter Priority: ";
                cin >> priority;
                queue.enqueue(name, priority);
                cout << "Patient Added Successfully" << endl;
                break;
            case 2:
                priority = queue.get_priority();
                name = queue.dequeue();
                cout << "Patient '" << name << "' with
priority '" << priority
                << "' removed" << endl;
                break;
            case 3:
                queue.display();
                break;
            case 4:
                cout << "Thank You" << endl;
                break;
            default:
                cout << "Invalid Choice" << endl;
                break;
        }
    } while (ch != 4);
    return 0;
}

```

Output:

Priorities are 0: General Checkup, 1: Non-Serious, 2: Serious

-----: MENU :-----

1. Add Patient
2. Remove Patient
3. Display Queue
4. Exit

Enter your choice: 1

Enter Patient Name: khushal

Enter Priority: 0

Patient Added Successfully

-----: MENU :-----

1. Add Patient
2. Remove Patient
3. Display Queue
4. Exit

Enter your choice: 1

Enter Patient Name: yash

Enter Priority: 2

Patient Added Successfully

-----: MENU :-----

1. Add Patient
2. Remove Patient
3. Display Queue
4. Exit

Enter your choice: 1

Enter Patient Name: ajay

Enter Priority: 1

Patient Added Successfully

-----: MENU :-----

1. Add Patient
2. Remove Patient
3. Display Queue
4. Exit

Enter your choice: 3

Queue is:

2 : yash

1 : ajay

0 : khushal

-----: MENU :-----

1. Add Patient
2. Remove Patient
3. Display Queue
4. Exit

Enter your choice: 2

Patient 'yash' with priority '2' removed

-----: MENU :-----

1. Add Patient
2. Remove Patient
3. Display Queue
4. Exit

Enter your choice: 3

Queue is:

1 : ajay

0 : khushal