```
#include <iostream>
using namespace std;
#define SIZE 5
class JobQueue {
private:
int queue[SIZE];
int front, rear;
public:
JobQueue() {
front = rear =-1;
}
void enQueue(int value) {
if (rear == SIZE- 1) {
cout << "Queue is FULL!!! Insertion is not possible!!!" << endl;
return;
}
if (front ==-1) {
front = 0;
}
rear++;
queue[rear] = value;
cout << "Job " << value << " added to the queue." << endl;
}
void deQueue() {
if (front ==-1 || front > rear) {
cout << "Queue is EMPTY!!! Deletion is not possible!!!" << endl;
return;
}
cout << "Job " << queue[front] << " deleted from the queue." << endl;
front++;
if (front > rear) {
front = rear =-1;
}
}
```

```
void display() {
if (front ==-1) {
cout << "Queue is EMPTY!!!" << endl;
return;
}
cout << "Jobs in the queue: ";
for (int i = front; i <= rear; i++) {
cout << queue[i] << " ";
}
cout << endl;
}
};
int main() {
JobQueue jq;
int choice, value;
do {
cout << "\nJob Queue Operations Menu:\n";
cout << "1. Add Job (enqueue)\n";
cout << "2. Delete Job (dequeue)\n";
cout << "3. Display Jobs\n";
cout << "4. Exit\n";
cout << "Select an option: ";
cin >> choice;
switch (choice) {
case 1:
cout << "Enter job number to add: ";
cin >> value;
jq.enQueue(value);
break;
case 2:
jq.deQueue();
break;
case 3:
jq.display();
break;
case 4:
cout << "Exiting the program." << endl;
break;
default:
cout << "Invalid option! Please try again." << endl;
}
} while (choice != 4);
return 0;
}
```