# Problem Set 2: Priority Queue

## An A&E Scheduler

As anyone who has been to a busy hospital A&E knows, people must wait for service. Although everyone might appear to be waiting in the same place, they are actually in separate groups and scheduled according to the seriousness of their condition.

This problem set develops a program that performs this scheduling with a priority queue.

Write a program that allows a medical officer to schedule treatments for patients coming into a hospital's A&E. Assume that, because some patients are in more critical condition than others, patients are not treated on a strictly first-come, first-served basis, but are assigned a priority when admitted. Patients with a high priority receive attention before those with a lower priority.

Patients come into the A&E in one of three conditions. In order of priority, the conditions are ranked as follows:

1. Critical
2. Serious
3. Minor

When the user selects the Schedule option, the program allows the user to enter a patient's name and condition, and the patient is placed in line for treatment according to the severity of his condition. When the user selects the Treat Next Patient option, the program removes and displays the patient first in line with the most serious condition.

When the user selects the Treat All Patients option, the program removes and displays all patients in order from patient to serve first to patient to serve last.

Each command button produces an appropriate message in the output area. The table below shows the interface's responses to the commands.

Commands of the Emergency Room Program

| User Command | Program Response |
|---|---|
| Schedule | Prompts the user for the patient's name and condition, and then prints <patient name> is added to the <condition> list. |
| Treat Next Patient | Prints <patient name> is being treated. |
| Treat All Patients | Prints <patient name> is being treated. … Prints <patient name> is being treated. |

Here is an interaction with the terminal-based interface:

```
Main menu
1 Schedule a patient
2 Treat the next patient
3 Treat all patients
4 Exit the program

Enter a number [1-4]: 1

Enter the patient's name: Bill
Patient's condition:
1 Critical
2 Serious
3 Fair

Enter a number [1-3]: 1
Bill is added to the critical list.

Main menu
1 Schedule a patient
2 Treat the next patient
3 Treat all patients
4 Exit the program

Enter a number [1-4]: 3

Bill / critical is being treated.
Martin / serious is being treated.
Ken / fair is being treated.
No patients available to treat.
```

You are advised to submit work that adopts good programming styles (e.g. modularity, documentation, error checking etc).