

```

import java.io.File;
import java.io.FileNotFoundException;
import java.util.Scanner;

// Author Name: Jarrett Crump
// Email: jarrett.crump@okstate.edu
// Date: 11/06/2022
// Program Description: Implements a priority queue using a heap structure
to create a basic healthcare database based around UNOS statues for organ
transplants

class App {
    public static void main(String[] args) throws Exception {
        Scanner inputScnr = new Scanner(System.in); // Declare a scanner
        Heap heap = new Heap(); // Declare the main heap structure
        displayMenu(inputScnr, heap); // Call the function to display the
menu for the first time
    }

    public static void displayMenu(Scanner inputScnr, Heap heap) { //
Displays the menu, and recursively calls itself until the program is
exited
        String choice;

System.out.println("\n-----
-----");

        System.out.println("WHAT WOULD YOU LIKE TO DO? (input only a
single number)");

        System.out.println("1. Import data");
        System.out.println("2. Peek");
        System.out.println("3. Next patient");
        System.out.println("4. Remove patient");
        System.out.println("5. Size");
        System.out.println("6. Update priority");
        System.out.println("7. Exit\n");
        choice = inputScnr.nextLine();

        switch (choice) {
            case "1": // input

```

```

        if (!heap.getRead()) {
            input(heap);
        }
        displayMenu(inputScnr, heap);
        break;
case "2": // peek
    if (heap.getRead()) {
        peek(heap);
    }
    displayMenu(inputScnr, heap);
    break;
case "3": // nextPatient
    if (heap.getRead()) {
        nextPatient(heap);
    }
    displayMenu(inputScnr, heap);
    break;
case "4": // removePatient
    if (heap.getRead()) {
        removePatient(heap, inputScnr);
    }
    displayMenu(inputScnr, heap);
    break;
case "5": // size
    size(heap);
    displayMenu(inputScnr, heap);
    break;
case "6": // updatePriority
    if (heap.getRead()) {
        updatePriority(heap, inputScnr);
    }
    displayMenu(inputScnr, heap);
    break;
case "7": // exit
    System.out.println("Exiting...");
    break;
default:
    System.out.println("Please input a valid option");
    displayMenu(inputScnr, heap);
}

```

```

    }

    private static void input(Heap heap) { // Inputs the datapoints into
the heap structure
        File file = new File("inputFile.txt"); // Finds the file at
.\inputFile.txt
        Scanner fileScnr = null;
        try {
            fileScnr = new Scanner(file);
            fileScnr.nextLine(); //Skip first line of file
        } catch (FileNotFoundException e) {
            System.out.println("Input file not found.");
            System.exit(1);
        }
        while (fileScnr.hasNextLine()) {
            String arr[] = fileScnr.nextLine().split(";");
            heap.add(new Entry(arr[0], arr[1], arr[2], arr[3], arr[4],
arr[5], arr[6], arr[7], arr[8], arr[9], arr[10], arr[11], arr[12])); //
Adds item to the heap
        }
        System.out.println("Input file is read sucessfully");
        heap.setRead(true); // Prevents data from being read multiple
times
    }

    private static void peek(Heap heap) { // Prints the info of the
highest priority patient without removing them from the heap
        Entry entry = heap.peek();
        System.out.println("The patient detail with the highest priority
is as follows:");
        System.out.println("Patient's first name: " +
entry.getPatient().getFirstName());
        System.out.println("Patient's last name: " +
entry.getPatient().getLastName());
        System.out.println("Patient's date of birth: " +
entry.getPatient().getDob());
        System.out.println("Address: " + entry.getPatient().getAddress());
        System.out.println("City: " + entry.getPatient().getCity());
        System.out.println("County: " + entry.getPatient().getCounty());
        System.out.println("State: " + entry.getPatient().getState());
    }

```

```

        System.out.println("ZIP Code: " + entry.getPatient().getZip());
        System.out.println("Phone number (1st preference): " +
entry.getPatient().getPhone1());
        System.out.println("Phone number (2nd preference): " +
entry.getPatient().getPhone2());
        System.out.println("Email Address: " +
entry.getPatient().getEmail());
        System.out.println("UNOS Status: " +
entry.getPatient().getUnosStatus());
        System.out.println("USNO Status updates:");
        for (int i = 0; i < entry.getPatient().getPastStatus().length;
i++) { // Prints all unos status changes
            if (entry.getPatient().getPastStatus()[i] != null) {
                System.out.println("Status changed from: " +
entry.getPatient().getPastStatus()[i] + " on: " +
entry.getPatient().getPastStatusDates()[i]);
            }
        }
    }

    private static void nextPatient(Heap heap) { // Prints the info of the
highest priority patient and removes them from the heap
        Entry entry = heap.next();
        System.out.println("The patient detail with the highest priority
is as follows:");
        System.out.println("Patient's first name: " +
entry.getPatient().getFirstName());
        System.out.println("Patient's last name: " +
entry.getPatient().getLastName());
        System.out.println("Patient's date of birth: " +
entry.getPatient().getDob());
        System.out.println("Address: " + entry.getPatient().getAddress());
        System.out.println("City: " + entry.getPatient().getCity());
        System.out.println("County: " + entry.getPatient().getCounty());
        System.out.println("State: " + entry.getPatient().getState());
        System.out.println("ZIP Code: " + entry.getPatient().getZip());
        System.out.println("Phone number (1st preference): " +
entry.getPatient().getPhone1());
        System.out.println("Phone number (2nd preference): " +
entry.getPatient().getPhone2());

```

```

        System.out.println("Email Address: " +
entry.getPatient().getEmail());
        System.out.println("UNOS Status: " +
entry.getPatient().getUnosStatus());
        System.out.println("USNO Status updates:");
        for (int i = 0; i < entry.getPatient().getPastStatus().length;
i++) {
            if (entry.getPatient().getPastStatus()[i] != null) {
                System.out.println("Status changed from: " +
entry.getPatient().getPastStatus()[i] + " on: " +
entry.getPatient().getPastStatusDates()[i]);
            }
        }
    }

    private static void removePatient(Heap heap, Scanner inputScnr) { //
Removes the specified patient regardless of their priority within the
queue
        System.out.println("Please enter the info of the patient to be
removed from the queue.");
        System.out.println("Please enter patient's first name: ");
        String firstName = inputScnr.nextLine();
        System.out.println("Please enter patient's last name: ");
        String lastName = inputScnr.nextLine();
        System.out.println("Please enter patient's date of birth: ");
        String dob = inputScnr.nextLine();
        System.out.println("Please enter patient's Address: ");
        String address = inputScnr.nextLine();
        System.out.println("Please enter patient's City: ");
        String city = inputScnr.nextLine();
        System.out.println("Please enter patient's County: ");
        String county = inputScnr.nextLine();
        System.out.println("Please enter patient's State: ");
        String state = inputScnr.nextLine();
        System.out.println("Please enter patient's ZIP Code: ");
        String zip = inputScnr.nextLine();
        System.out.println("Please enter patient's Phone number (1st
preference): ");
        String phone1 = inputScnr.nextLine();

```

```

        System.out.println("Please enter patient's Phone number (2nd
preference): ");
        String phone2 = inputScnr.nextLine();
        System.out.println("Please enter patient's Email Address: ");
        String email = inputScnr.nextLine();
        System.out.println("Please enter patient's UNOS Status: ");
        String unosStatus = inputScnr.nextLine();
        int index = heap.contains(new Patient(firstName, lastName,
address, city, county, state, zip, phone1, phone2, email, unosStatus,
dob));
        if (index > 0) {
            heap.remove(index);
            System.out.println("The requested patient's record has been
removed from the queue.");
        } else {
            System.out.println("The requested patient's record is not
found.");
        }
    }

    private static void size(Heap heap) { // Returns the number of entries
in the database
        System.out.println("Number of records in the database: " +
heap.getSize());
    }

    private static void updatePriority(Heap heap, Scanner inputScnr) { //
Updates the priority of the specified entry in the database
        System.out.println("Please enter the info of the patient to be
updated.");
        System.out.println("Please enter patient's first name: ");
        String firstName = inputScnr.nextLine();
        System.out.println("Please enter patient's last name: ");
        String lastName = inputScnr.nextLine();
        System.out.println("Please enter patient's date of birth: ");
        String dob = inputScnr.nextLine();
        System.out.println("Please enter patient's Address: ");
        String address = inputScnr.nextLine();
        System.out.println("Please enter patient's City: ");
        String city = inputScnr.nextLine();
    }

```

```

        System.out.println("Please enter patient's County: ");
        String county = inputScnr.nextLine();
        System.out.println("Please enter patient's State: ");
        String state = inputScnr.nextLine();
        System.out.println("Please enter patient's ZIP Code: ");
        String zip = inputScnr.nextLine();
        System.out.println("Please enter patient's Phone number (1st
preference): ");
        String phone1 = inputScnr.nextLine();
        System.out.println("Please enter patient's Phone number (2nd
preference): ");
        String phone2 = inputScnr.nextLine();
        System.out.println("Please enter patient's Email Address: ");
        String email = inputScnr.nextLine();
        System.out.println("Please enter patient's old UNOS Status: ");
        String unosStatus = inputScnr.nextLine();
        int index = heap.contains(new Patient(firstName, lastName,
address, city, county, state, zip, phone1, phone2, email, state,
unosStatus, dob));
        if (index > 0) {
            System.out.println("Patient found, enter new UNOS status:");
            unosStatus = inputScnr.nextLine();
            Entry entry = heap.getEntry(index);
            heap.updatePriority(index, unosStatus);
            System.out.println("The requested patient's record has been
updated.");
            System.out.println("Patient's first name: " +
entry.getPatient().getFirstName());
            System.out.println("Patient's last name: " +
entry.getPatient().getLastName());
            System.out.println("Patient's date of birth: " +
entry.getPatient().getDob());
            System.out.println("Address: " +
entry.getPatient().getAddress());
            System.out.println("City: " + entry.getPatient().getCity());
            System.out.println("County: " +
entry.getPatient().getCounty());
            System.out.println("State: " + entry.getPatient().getState());
            System.out.println("ZIP Code: " +
entry.getPatient().getZip());

```

```
        System.out.println("Phone number (1st preference): " +
entry.getPatient().getPhone1());
        System.out.println("Phone number (2nd preference): " +
entry.getPatient().getPhone2());
        System.out.println("Email Address: " +
entry.getPatient().getEmail());
        System.out.println("UNOS Status: " +
entry.getPatient().getUnosStatus());
        System.out.println("USNO Status updates:");
        for (int i = 0; i < entry.getPatient().getPastStatus().length;
i++) {
            if (entry.getPatient().getPastStatus()[i] != null) {
                System.out.println("Status changed from: " +
entry.getPatient().getPastStatus()[i] + " on: " +
entry.getPatient().getPastStatusDates()[i]);
            }
        }
    } else {
        System.out.println("The requested patient's record is not
found.");
    }
}
}
```