Data Structures and Algorithms

HEALTHCARE MANAGEMENT SYSTEM

Course Project Report

School of Computer Science and Engineering

2024-25

Contents

1. Course and Team Details

1.1 Course details

Course Name	Design and Analysis of Algorithms
Course Code	24ECSC205
Semester	III
Division	A
Year	2024-25
Instructor	Mallikarjun Akki

1.2 Team Details

Si. No.	Roll No.	Name
1.	118	Prasad Morab
2.	120	Prasanna Anagal
3.	137	Siddharth Prabhu
4.	140	Shashank Joshi

1.3 Report Owner

Roll No.	Name
120	Prasanna Anagal

2. Introduction

The Healthcare Management System (HMS) is a project that uses efficient algorithms and data structures to optimize healthcare operations. In a city, health of citizens matters the most. So this project is picked although there were many problems that may be addressed. HMS addresses four key areas:
Emergency Incident Categorization – categorizing and prioritizing medical emergencies;
Patient Appointment Scheduling – scheduling appointments for patients in an efficient manner;
Medical Supply Inventory Management – tracking and restocking of medical supplies;
Healthcare Resource Allocation – optimizing the critical distribution of healthcare resources.
HMS is done with the help of algorithms of Dijkstra's Algorithm, the Rabin-Karp string matching, QuickSort, Depth First Search, and Priority Queues, BST, and Union Find to bring improved efficiency in timely real-time decision making.

It is based on the principles from algorithmic problem-solving and data structures, discussed in key references like Data Structures and Algorithm Analysis in C++ by Weiss [1] and Introduction to Algorithms by Cormen et al. [2]. This helped choose suitable algorithms for every module, scalable enough for the healthcare scenario, while being very efficient in practice.

3. Problem Statement

3.1 Domain

The HMS automates healthcare operations such as incident management, patient scheduling, resource allocation and inventory management. These tasks are full of inefficiencies and delays, leading to poor service quality and increased costs. The system's objective is to resolve these challenges by providing an algorithm-driven, scalable, and user-friendly solution.

3.2 Module Description

The **Patient Appointment Scheduling System** is designed to optimize appointment scheduling by reducing wait times and prioritizing urgent cases.
Key Algorithms Used:

1. **Min-Heap (Priority Queue)** – Manages urgent appointments efficiently.
2. **Queue (FIFO)** – Ensures fair processing of regular appointments.
3. **Bubble Sort** – Sorts available time slots for better scheduling.

4. Functionality Selection

Si. No.	Functionality Name	Known	Unknown	Principles applicable	Algorithms	Data Structures
	Name the functionality within the module	What information do you already know about the module? What kind of data you already have? How much of process information is known?	What are the pain points? What information needs to be explored and understood? What are challenges?	What are the supporting principles and design techniques?	List all the algorithms you will use	What are the supporting data structures?
1	Urgent Appointment Scheduling	Some appointments are time-sensitive and must be prioritized.	Need a system that automatically prioritizes urgent cases without affecting fairness.	Greedy Approach	Min-Heap (Priority Queue)	Heaps
2	Regular Appointment Management	Most appointments follow a First-In-First-Out (FIFO) structure.	Avoiding overlaps and long wait times while ensuring fairness in scheduling.	Queue Processing	Queue (FIFO)	Queues
3	Sorting Available Time Slots	Appointments should be allocated to the earliest available slot.	Sorting needs to be quick and efficient, as new slots open dynamically.	Iterative Approach	Bubble Sort	Arrays

5. Functionality Analysis

5.1 Urgent Appointment Scheduling (Min-Heap)

- Workflow:
    1. Appointments are inserted into a Min-Heap based on urgency.
    2. The highest-priority (most urgent) patient is always at the top.
    3. Ensures urgent patients are treated first.
- Efficiency:
    - Time Complexity: O(log n) for insertion and deletion.
    - Space Complexity: O(n) for storing all patients in a heap.

5.2 Regular Appointment Management (Queue - FIFO)

- Workflow:
    1. Regular appointments are processed in a queue.
    2. Ensures fairness by following First-In-First-Out.
    3. Prevents starvation of lower-priority patients.
- Efficiency:
    - Time Complexity: O(1) for insertion and removal.
    - Space Complexity: O(n) for maintaining the queue.

5.3 Sorting Available Time Slots (Bubble Sort)

- Workflow:
    1. Available time slots are sorted using Bubble Sort.
    2. Ensures patients get the earliest available slot.
- Efficiency:
    - Time Complexity: O(n^2) in the worst case.
    - Space Complexity: O(1).

6. Conclusion

The Patient Appointment Scheduling System improves efficiency and fairness in scheduling by ensuring that urgent patients receive priority while maintaining an orderly queue for general appointments. The use of a Min-Heap (priority queue) for urgent cases, FIFO queue for regular patients, and Bubble Sort for optimizing time slots provides a balanced and structured approach to appointment management.
This module demonstrates the power of priority queues in healthcare scheduling, ensuring efficient and timely patient care. It also emphasizes the significance of sorting algorithms in optimizing available slots. Overall, this system contributes to a more effective and accessible healthcare scheduling process.

7. References

[1] **Weiss, Mark Allen.** *Data Structures and Algorithm Analysis in C++ (4th Edition).* Pearson, 2013.
[2] **Thomas H. Cormen, Clifford Stein, Ronald L. Rivest, and Charles E. Leiserson.** *Introduction to Algorithms (3rd ed.).* The MIT Press, 2009.

~*~*~*~*~*~*~*~