1. Introduction

In the healthcare domain, efficient management of patient records and medical resources is essential for delivering timely and accurate care. Manual systems are prone to errors, time delays, and redundancy. To mitigate such issues, computerized systems have become standard in modern hospitals.

This project presents a **console-based Hospital Management System** written entirely in **C**, using structured programming techniques and fundamental data structures. The system facilitates the registration of patients, maintains doctor records, suggests appropriate specialists based on symptoms, and allows for patient discharge operations.

The project serves as an excellent example of **modular design**, **structure-based data modeling**, and **console I/O programming**, suitable for undergraduate curriculum or practical learning.

2. Objectives

- Design a hospital management system using structured C programming.
- Implement patient registration with disease-based doctor suggestion logic.
- Display and manage patient and doctor records using memory-resident data.
- Handle patient discharges dynamically (removal from memory).
- Demonstrate technical concepts: struct, array of structs, modular functions, string handling, input/output formatting.

3. Methodology

The system is designed using a **modular programming approach** with the following steps:

- **Data Structures Used**:
  Two struct types are defined — Patient and Doctor. Arrays store data for up to 100 patients and 10 doctors.
- **Functional Design**:
  - addDoctors() : initializes the doctor database.
  - addPatient() : takes patient details, assigns an ID, stores them, and suggests a doctor.
  - listPatients() : displays all registered patients.
  - dischargePatient() : removes a patient by ID.
  - listDoctors() : shows available doctors.
- **Decision Logic**:
  - Doctor suggestions are based on substring matching of the disease description with known specializations.