TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TPHCM

MÔN HỌC: OBJECT-ORIENTED PROGRAMMING

FROM REQUIREMENTS TO OOP MODELS: SMALL CLINIC MANAGEMENT SYSTEM

Sinh viên thực hiện:

Trần Gia Kiệt: 24110103

Thành phố Hồ Chí Minh, tháng 9 năm 2025

1. Object-Oriented Analysis (OOA)

Step 1: Identify Objects

The main objects in the system are:

1. **Person** – Base class for any human in the clinic.
2. **Patient** – Represents general patients.
3. **ChronicPatient** – Specialized patient with an ongoing medical condition.
4. **Doctor** – Represents doctors in the clinic.
5. **Appointment** – Represents a scheduled visit between a patient and a doctor.

Step 2: Attributes for Each Object

Object	**Attributes**
Person	name, ID, age, gender
Patient	address, healthInsuranceID, nextAppointment, appointment history
ChronicPatient	conditionType, lastCheckup (inherits Patient)
Doctor	address, healthInsuranceID, medicalSpecialty, list of appointments
Appointment	doctorName, patientName, date, time, reason, status

Step 3: Methods

Object	**Methods**
Person	displayInfo()
Patient	scheduleAppointment(), addAppointment(), displayInfo()
ChronicPatient	scheduleAppointment() (override), displayInfo() (override)
Doctor	addAppointment(), displayInfo()
Appointment	display(), cancel(), complete(), getters for date, time, status

Step 4: Inheritance Relationships

- Person is the base class.
- Patient inherits from Person.
- ChronicPatient inherits from Patient and overrides specific methods.
- Doctor inherits from Person.

2. Class Design Explanation

- **Person**: Base class for common attributes (name, ID, age, gender) and displayInfo().
- **Patient**: Adds clinic-specific attributes (address, insurance, next appointment). Manages a list of appointments.
- **ChronicPatient**: Inherits from Patient. Adds conditionType and lastCheckup. Overrides scheduleAppointment() to handle more frequent appointments.
- **Doctor**: Inherits from Person. Tracks assigned appointments. Can display all upcoming appointments.
- **Appointment**: Encapsulates all information about a visit: doctor, patient, date, time, reason, and status. Uses getters to enforce encapsulation.

Key OOP Principles Used:

1. **Encapsulation**: Attributes are protected or private; getters and setters used where needed.
2. **Inheritance**: ChronicPatient extends Patient, Person as the base.
3. **Polymorphism**: displayInfo() is virtual, allowing derived classes to override behavior.
4. **Composition**: Patients and Doctors maintain Appointment objects.

3. Code Walkthrough

a) Person Class

```
class Person {
protected:
   string name, id, gender;
   int age;
public:
   virtual void displayInfo();
};
```

- Base class for both patients and doctors.
- displayInfo() is virtual for polymorphic behavior.

b) Patient & ChronicPatient

```
class Patient : public Person {
   string address, healthInsuranceID, nextAppointment;
   vector<Appointment> appHistory;
   virtual void scheduleAppointment(...);
};

class ChronicPatient: public Patient {
   string conditionType, lastCheckup;
   void scheduleAppointment(...) override;
};
```

- Patients can schedule appointments and store history.
- ChronicPatient overrides scheduleAppointment() to ensure more frequent follow-ups.

c) Doctor

```
class Doctor: public Person {
   string address, healthInsuranceID, medicalSpecialty;
   vector<Appointment> appointments;
};
```

- Manages a list of assigned appointments.
- displayInfo() shows upcoming appointments.

d) Appointment

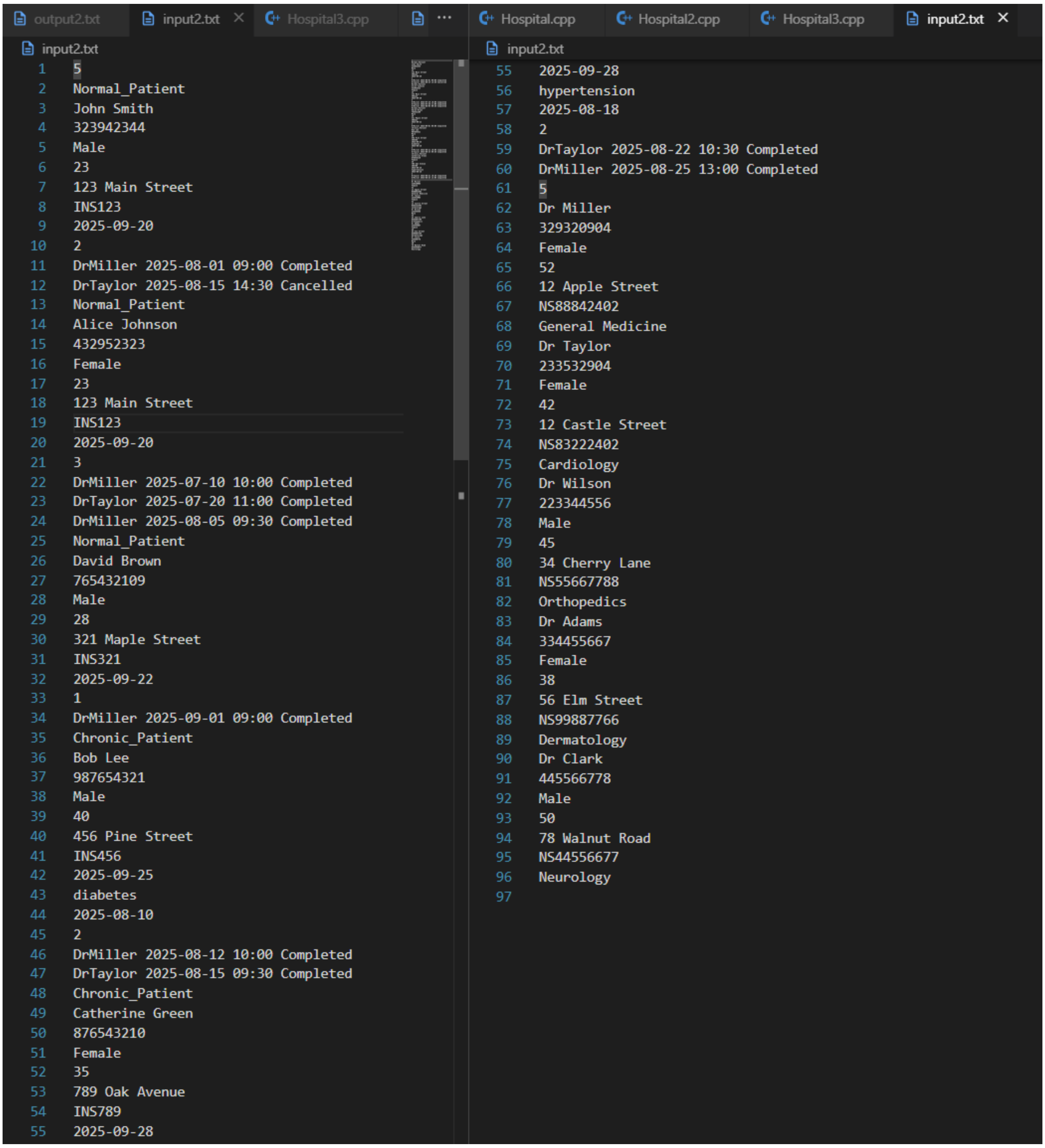
```
class Appointment {
protected:
   string doctorName, patientName, date, time, reason, status;
public:
   void display();
   void cancel();
   void complete();
   string getDate(), getTime(), getStatus();
};
```

- Encapsulates visit details.
- Provides getters to allow access without exposing internal state.

e) Main Function

- Reads **input2.txt**: 5 patients (3 normal, 2 chronic) and 5 doctors.
- Adds appointments for patients and doctors.
- Displays all patients and doctors to **output2.txt**.

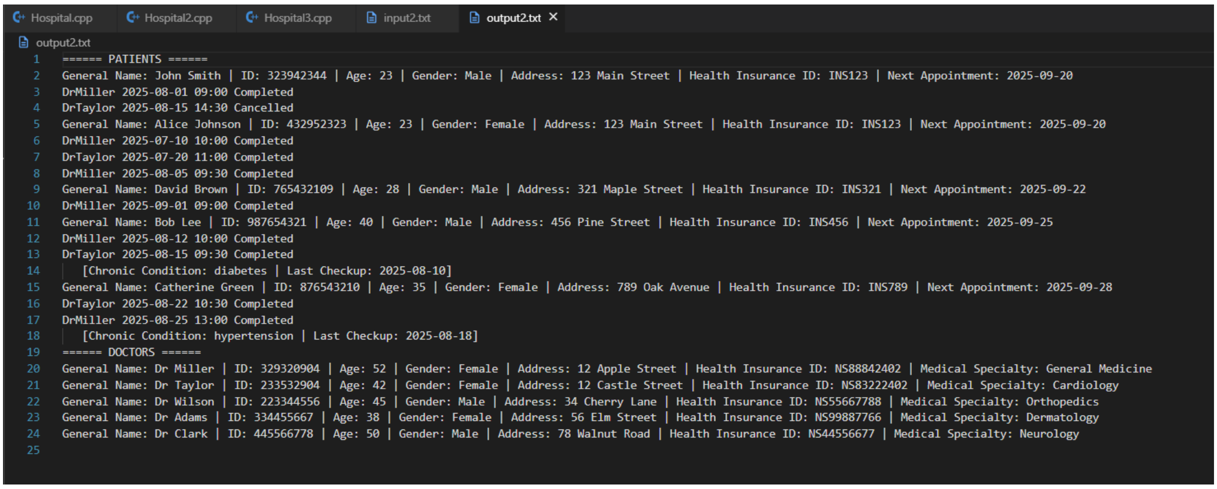
Sample Input2.txt



```
5
Normal_Patient
John Smith
323942344
Male
23
123 Main Street
INS123
2025-09-20
2
DrMiller 2025-08-01 09:00 Completed
DrTaylor 2025-08-15 14:30 Cancelled
Normal_Patient
Alice Johnson
432952323
Female
23
123 Main Street
INS123
2025-09-20
3
DrMiller 2025-07-10 10:00 Completed
DrTaylor 2025-07-20 11:00 Completed
DrMiller 2025-08-05 09:30 Completed
Normal_Patient
David Brown
765432109
Male
28
321 Maple Street
INS321
2025-09-22
1
DrMiller 2025-09-01 09:00 Completed
Chronic_Patient
Bob Lee
987654321
Male
40
456 Pine Street
INS456
2025-09-25
diabetes
2025-08-10
2
DrMiller 2025-08-12 10:00 Completed
DrTaylor 2025-08-15 09:30 Completed
Chronic_Patient
Catherine Green
876543210
Female
35
789 Oak Avenue
INS789
2025-09-28
hypertension
2025-08-18
2
DrTaylor 2025-08-22 10:30 Completed
DrMiller 2025-08-25 13:00 Completed
5
Dr Miller
329320904
Female
52
12 Apple Street
NS88842402
General Medicine
Dr Taylor
233532904
Female
42
12 Castle Street
NS83222402
Cardiology
Dr Wilson
223344556
Male
45
34 Cherry Lane
NS55667788
Orthopedics
Dr Adams
334455667
Female
38
56 Elm Street
NS99887766
Dermatology
Dr Clark
445566778
Male
50
78 Walnut Road
NS44556677
Neurology

```

Sample Output2.txt



```
====== PATIENTS ======
General Name: John Smith | ID: 323942344 | Age: 23 | Gender: Male | Address: 123 Main Street | Health Insurance ID: INS123 | Next Appointment: 2025-09-20
DrMiller 2025-08-01 09:00 Completed
DrTaylor 2025-08-15 14:30 Cancelled
General Name: Alice Johnson | ID: 432952323 | Age: 23 | Gender: Female | Address: 123 Main Street | Health Insurance ID: INS123 | Next Appointment: 2025-09-20
DrMiller 2025-07-10 10:00 Completed
DrTaylor 2025-07-20 11:00 Completed
DrMiller 2025-08-05 09:30 Completed
General Name: David Brown | ID: 765432109 | Age: 28 | Gender: Male | Address: 321 Maple Street | Health Insurance ID: INS321 | Next Appointment: 2025-09-22
DrMiller 2025-09-01 09:00 Completed
General Name: Bob Lee | ID: 987654321 | Age: 40 | Gender: Male | Address: 456 Pine Street | Health Insurance ID: INS456 | Next Appointment: 2025-09-25
DrMiller 2025-08-12 10:00 Completed
DrTaylor 2025-08-15 09:30 Completed
   [Chronic Condition: diabetes | Last Checkup: 2025-08-10]
General Name: Catherine Green | ID: 876543210 | Age: 35 | Gender: Female | Address: 789 Oak Avenue | Health Insurance ID: INS789 | Next Appointment: 2025-09-28
DrTaylor 2025-08-22 10:30 Completed
DrMiller 2025-08-25 13:00 Completed
   [Chronic Condition: hypertension | Last Checkup: 2025-08-18]
====== DOCTORS ======
General Name: Dr Miller | ID: 329320904 | Age: 52 | Gender: Female | Address: 12 Apple Street | Health Insurance ID: NS88842402 | Medical Specialty: General Medicine
General Name: Dr Taylor | ID: 233532904 | Age: 42 | Gender: Female | Address: 12 Castle Street | Health Insurance ID: NS83222402 | Medical Specialty: Cardiology
General Name: Dr Wilson | ID: 223344556 | Age: 45 | Gender: Male | Address: 34 Cherry Lane | Health Insurance ID: NS55667788 | Medical Specialty: Orthopedics
General Name: Dr Adams | ID: 334455667 | Age: 38 | Gender: Female | Address: 56 Elm Street | Health Insurance ID: NS99887766 | Medical Specialty: Dermatology
General Name: Dr Clark | ID: 445566778 | Age: 50 | Gender: Male | Address: 78 Walnut Road | Health Insurance ID: NS44556677 | Medical Specialty: Neurology
```

- Output shows patient details, appointment history, doctor details, and doctor-assigned appointments.
- Confirms **inheritance, polymorphism, and encapsulation** are functioning.

---

4. LLM Usage

- **Tool Used**: ChatGPT
- **Purpose**:
    - Brainstorm class design and attributes for the clinic system.
    - Suggest method names and inheritance patterns.
- **Example Prompts**:

1. "Suggest classes and methods for a small clinic management system in C++"
2. "How to model chronic patients using inheritance in C++"

Appendix: ChatGPT Responses

- Advised using inheritance for chronic patients.
- Recommended virtual methods for polymorphic behavior (displayInfo()).
- Suggested encapsulating appointment attributes and using getters.
- Suggested the test cases and output

---

5. Conclusion

- Successfully implemented a **Small Clinic Management System** in C++.
- Demonstrated OOP concepts: classes, inheritance, polymorphism, encapsulation.
- Included test cases and sample input/output.
- Ethical use of an LLM to assist in **conceptual design** only.