

## Lab 5: Stack, Queue and inheritance

### Bài 1:

#### Student Management System Assignment

##### Student Classes

- Store basic student information (id, name, age, gpa)
- Provide appropriate constructor and string representation methods
- Include additional attributes: country, language score
- Override string representation to include the additional attributes
- Include additional attributes: research area, thesis title
- Override string representation to include the additional attributes

##### Stack /Queue Class

- Implement a basic stack/queue data structure with the following methods:
- push: Add an item to the stack/ queue
- pop: Remove and return the top item
- is\_empty: Check if the stack/queue is empty
- show: Display stack/queue contents

##### StudentManagement Class

- Use the Stack/Queue class to manage student objects
- Implement the following functionality:

##### Required Functionality

- Add different types of students (regular, international, graduate)
- Delete students
- Display student list
- Search students by ID
- Update student information
- Find the oldest student
- Find student(s) with the highest GPA
- Sort students by GPA using Bubble Sort
- Sort students by ID using Selection Sort
- Save student data to a file
- Read student data from a file

### Bài 2: Healthcare Management System Assignment

#### Healthcare Professional Classes

- Store basic healthcare professional information (id, name, age, department)
- Provide appropriate constructor and string representation methods

##### Doctor Class (extends HealthcareProfessional)

- Include additional attributes: specialization, years\_experience
- Override string representation to include the additional attributes

##### Nurse Class (extends HealthcareProfessional)

- Include additional attributes: certification, shift
- Override string representation to include the additional attributes

##### Stack /Queue Class

- Implement a basic stack/queue data structure with the following methods:
- push: Add an item to the stack/ queue
- pop: Remove and return the top item
- is\_empty: Check if the stack/ queue is empty
- show: Display stack/queue contents

##### HealthcareManagement Class

- Use the Stack/Queue class to manage healthcare professional objects

- Implement the following functionality:

**Required Functionality**

- Add different types of healthcare professionals (regular, doctor, nurse)
- Delete professionals
- Display professional list
- Search professionals by ID
- Update professional information
- Find the most experienced doctor
- Find the oldest healthcare professional
- Sort doctors by years of experience using Bubble Sort
- Sort professionals by ID using Selection Sort
- Count professionals by department
- Save professional data to a file
- Read professional data from a file