**HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY**

**SCHOOL OF SCIENCE AND INFORMATION TECHNOLOGY**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***



**MIDTERM PROJECT REPORT**

**SUBJECT: Object-Oriented Language and Theory (Java)**

***Topic: An application to demonstrate some data structures***

**Lecturer: Dr. Nguyễn Thanh Hùng**

**Class ID: 122152**

**Members:**

**Nguyễn Văn Tỉnh 20184313**

**Trần Lê Hải Dương 20184248**

**Đỗ Quang Huy 20184271**

**MỤC LỤC**

Chương 1: Khảo sát, đặc tả yêu cầu bài toán

Chương 2: Phân tích thiết kế bài toán

Chương 3: Công nghệ và thuật toán sử dụng

Chương 4: Xây dựng chương trình minh họa

Chương 5: Kết luận

## Chương 1: Đặc tả yêu cầu đề tài

**1.Mô tả yêu cầu đề tài**

Xây dựng một ứng dụng đồ họa thể hiện các cấu trúc dữ liệu:

* Linked list
* Stack
* Balanced tree
* AVL tree

Sử dụng ngôn ngữ lập trình hướng đối tượng java và javafx để xây dựng chương trình.

**2.Biểu đồ use case**

## 

## Chương 2: Phân tích, thiết kế bài toán

1. **Phân tích bài toán**

+Linked List: là một dãy các cấu trúc dữ liệu được kết nối với nhau thông qua các liên kết (link). Hiểu một cách đơn giản thì Danh sách liên kết là một cấu trúc dữ liệu bao gồm một nhóm các nút (node) tạo thành một chuỗi. Mỗi nút gồm dữ liệu ở nút đó và tham chiếu đến nút kế tiếp trong chuỗi.

+Stack: Một ngăn xếp là một cấu trúc dữ liệu dạng thùng chứa (container) của các phần tử (thường gọi là các nút (node)) và có hai phép toán cơ bản: *push* and *pop*. *Push* bổ sung một phần tử vào đỉnh (top) của ngăn xếp, nghĩa là sau các phần tử đã có trong ngăn xếp. *Pop* giải phóng và trả về phần tử đang đứng ở đỉnh của ngăn xếp. Trong stack, các đối tượng có thể được thêm vào stack bất kỳ lúc nào nhưng chỉ có đối tượng thêm vào sau cùng mới được phép lấy ra khỏi stack.

+AVL tree: là cây tìm kiếm nhị phân có độ cân bằng cao. Cây AVL kiểm tra độ cao của các cây con bên trái và cây con bên phải và bảo đảm rằng hiệu số giữa chúng là không lớn hơn 1. **Hiệu số** này được gọi là **Balance Factor (Nhân tố cân bằng)**. Khi thêm hoặc xóa 1 node thì cây sẽ có những action để đảm bảo Balance factor không lớn hơn 1.

+Balanced tree: là một cấu trúc dữ liệu dạng cây cho phép tìm kiếm, truy cập tuần tự, chèn, xóa trong thời gian lôgarit. B-cây là một tổng quát hóa của cây nhị phân tìm kiếm, trong đó một nút có thể có nhiều hơn hai con. Không như cây nhị phân tìm kiếm tự cân bằng, B-cây được tối ưu hóa cho các hệ thống đọc và ghi dữ liệu lớn. Nó thường được dùng trong các cơ sở dữ liệu và hệ thống tập tin.

1. **Thiết kế bài toán**

Biểu đồ lớp:

## Chương 3: Công nghệ và thuật toán sử dụng

- Các công nghệ sử dụng:

+ Eclipse Java

+ JavaFX

+ JavaFX SceneBuilder

## Chương 4: Chương trình minh họa

### Chương 5: Kết luận

#### 1. Những việc đã thực hiện

* Thực hiện xây dựng các cấu trúc dữ liệu: Stack, Linked List, AVL tree, Balanced tree
* Xây dựng ứng dụng minh họa các cấu trúc dữ liệu bằng hướng đối tượng
* Thiết kế giao diện với JavaFX

#### 2. Những điều thu được

* Thu được kiến thức, kinh nghiệm trong việc thiết kế 1 chương trình hướng đối tượng
* Học được cách sử dụng JavaFX cho việc thiết kế giao diện đồ họa