

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH  
TRƯỜNG ĐẠI HỌC BÁCH KHOA  
KHOA KHOA HỌC - KỸ THUẬT MÁY TÍNH



LẬP TRÌNH NÂNG CAO (MỞ RỘNG)

---

# Đề tài mở rộng OOP & FP

---

GVHD: Trương Tuấn Anh  
SV: Đỗ Huy Hoàng - 2013219

TP. HỒ CHÍ MINH, NGÀY 10 THÁNG 6 NĂM 2022

# Mục lục

<b>1</b>	<b>So sánh OOP trong Java với Ruby</b>	<b>2</b>
1.1	Sự giống nhau . . . . .	2
1.2	Sự khác nhau . . . . .	2
<b>2</b>	<b>So sánh FP trong Haskell với SML</b>	<b>2</b>
2.1	Sự giống nhau . . . . .	2
2.2	Sự khác nhau . . . . .	2
<b>3</b>	<b>Viết 1 game đơn giản bằng C++</b>	<b>2</b>
3.1	Hiện thực . . . . .	3
3.2	Các vấn đề gặp phải khi làm game . . . . .	3
3.3	Đánh giá . . . . .	3
3.4	Hướng phát triển . . . . .	3

## 1 So sánh OOP trong Java với Ruby

### 1.1 Sự giống nhau

- Đây là ngôn ngữ lập trình bậc cao, hướng đối tượng.

### 1.2 Sự khác nhau

- Java được compile thành bytecode rồi interpret để chạy, còn Ruby thì được interpret để chạy.
- Class trong Java có access modifier là private và public, còn Ruby thì không có mà mặc định là public. Do vậy Ruby có `private_constant` để che giấu đi class.
- Java là ngôn ngữ static typing, Ruby là ngôn ngữ dynamic typing.
- Java xem mọi item đều là object trừ primitive types (integers, floats, booleans), còn Ruby coi primitive types là object.
- Java không support operator overloading, còn Ruby thì có.

## 2 So sánh FP trong Haskell với SML

### 2.1 Sự giống nhau

- Đây có infinite lists.
- Đây có higher-order functions.
- Đây có pattern matching.

### 2.2 Sự khác nhau

- Haskell dùng cơ chế lazy evaluation, SML dùng eager evaluation.
- SML được interpret, còn Haskell thì compile rồi interpret.

## 3 Viết 1 game đơn giản bằng C++

Tựa game: Snake game

IDE: Visual Studio

Graphics Library: SFML C++

Hướng dẫn chơi:

- Dùng các phím mũi tên để đổi hướng con rắn
- Có thể pause/unpause game bằng nút Space
- Khi game over, nhấn nút Enter để chơi lại



### 3.1 Hiện thực

Project gồm các file:

- Snake.h, Snake.cpp quản lý snake bằng 1 vector gồm các snake sections.
- Food.h, Food.cpp quản lý Food cho Snake là Apple.
- Wall.h, Wall.cpp quản lý wall section
- Engine.h, Engine.cpp quản lý render window, deque lưu lại các direction do user nhập, time counter để xác định thời window refresh rate, snake, apple, wall (gồm các wall sections), game state (running, paused, game over), random generator (phục vụ việc sinh địa điểm đặt apple ngẫu nhiên).
- Main.cpp chạy game engine để bắt đầu game.

### 3.2 Các vấn đề gặp phải khi làm game

- Coordinate system với góc trên bên trái là (0,0) còn góc dưới bên phải là  $(x_{max}, y_{max})$ . => nên hướng đi lên và xuống của rắn sẽ bị đảo ngược.
- Loại trừ các trường hợp không cho rắn đi ngược chiều (ví dụ: khi rắn đi lên, người dùng nhấn mũi tên đi xuống sẽ không được thực hiện).
- Dùng deque để đảm bảo mỗi lần update window chỉ có nhiều nhất 1 lần đổi hướng cũng như lưu lại các hướng khi người dùng nhấn combo nhiều hướng.
- Chỉ khởi tạo trái táo 1 lần, khi con rắn ăn táo thì chỉ đổi vị trí quả táo, không cần xóa đi => tăng hiệu năng.

### 3.3 Đánh giá

- Game cơ bản đã chạy được
- Cần wrap wall section trong class wall và để engine quản lý class wall thay vì wall section.

### 3.4 Hướng phát triển

- Load các map với các kết cấu wall khác nhau
- Tạo class wall để tùy biến phát triển (ví dụ wall lúc ẩn lúc hiện).

## Tài liệu

- [1] Benjamin J. Keller. *Functional Programming Languages*. <https://courses.cs.vt.edu/cs3304/Spring02/lectures/lect04.pdf>.
- [2] GeeksforGeeks. *Difference Between Java And Ruby*. <https://www.geeksforgeeks.org/difference-between-java-and-ruby/>.
- [3] Wikipedia. *Java (programming language)*. [https://en.wikipedia.org/wiki/Java\\_\(programming\\_language\)](https://en.wikipedia.org/wiki/Java_(programming_language)).
- [4] Wikipedia. *Ruby (programming language)*. [https://en.wikipedia.org/wiki/Ruby\\_\(programming\\_language\)](https://en.wikipedia.org/wiki/Ruby_(programming_language)).



- [5] GeeksforGeeks. *Operator Overloading in Ruby*. <https://www.geeksforgeeks.org/operator-overloading-in-ruby/>.