# **Bài 09. Class, operator overloading**

**Mục tiêu:** *Luyện tập cú pháp class, cách sử dụng hằng thành viên, định nghĩa toán tử*

**Yêu cầu nộp bài** *: Toàn bộ code phần A: một chương trình cho câu A1 và chương trình thứ hai cho các câu còn lại. Nếu bạn làm đến A6, bạn sẽ tách chương trình thứ hai đó thành 3 file.*

*Các bài (\*) không bắt buộc, nếu làm bạn sẽ được bù điểm cho bài cũ.*

***Lưu ý: chép bài hoặc cho chép bài sẽ dẫn đến trượt môn học!***

## Thực hành

1. ***Cú pháp class.*** Với cài đặt danh sách liên kết đôi DList bạn đã làm ở bài trước (bổ sung để hoàn thiện nếu bài nộp trước chưa hoàn thiện). Hãy chuyển DList từ struct sang cấu trúc class, sao cho các biến thành viên đều ở dạng private, trong khi các hàm public. Hàm main và các hàm test không nên phải sửa đổi gì mà vẫn chạy như cũ.
2. ***Hằng thành viên.*** Viết lớp Fraction (phân số), trong đó mỗi fraction cần có hai biến thành viên numerator (tử số) và denominator (phân số), cả hai đều là hằng, public và được khởi tạo duy nhất một lần tại constructor. Viết thêm hàm thành viên print() với nhiệm vụ in ra màn hình dạng x/y. Hãy thử nghiệm trong hàm main(): tạo phân số và in ra màn hình.
3. ***Operator overload.*** Hãy định nghĩa các phép + - \*/ cho Fraction. Thử nghiệm các phép toán đó trong hàm main.
4. Bổ sung code để kết quả của mỗi phép toán trên đều là một phân số tối giản.
5. ***(\*) Assert.*** Tại mỗi hàm định nghĩa một phép toán, bạn hãy dùng assert ở đầu hàm để kiểm tra tiền điều kiện mẫu số phải khác 0 cho mỗi phép toán. Tại cuối mỗi hàm test, đặt các assert để kiểm tra xem phân số kết quả của phép toán có giá trị đúng như mong đợi.

6. ***Tách header và cài đặt class***. Tách Fraction thành các file fraction.h chứa khai báo class, faction.cpp chứa cài đặt các hàm thành viên. Nội dung chương trình thử nghiệm (chứa hàm main) đặt vào file fraction-demo.cpp.

Xem hướng dẫn nối include tại bài giảng.

Thử biên dịch demo với mã nguồn fraction.cpp bằng dòng lệnh g++ demo.cpp faction.cpp.

Thử biên dịch file fraction.cpp trước bằng lệnh g++ -c fraction.cpp, kết quả biên dịch là file fraction.o. Sau đó dịch demo.cpp và link nó với thư viện fraction đã biên dịch bằng lệnh g++ demo.cpp fraction.o.

## B. Câu hỏi trắc nghiệm.

<Thầy Khôi sẽ bổ sung vào đây>