

**TRƯỜNG ĐẠI HỌC GTVT TP HCM**  
**KHOA CNTT**

**MÔN HỌC LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG**  
**BÀI THỰC HÀNH**  
**CHƯƠNG 3: QUÁ TẢI TOÁN TỬ**

Cài đặt lớp số hữu tỉ (Rational) gồm:

- Các thành viên dữ liệu (**private**):
  - + Tử số (*numerator*) là số nguyên
  - + Mẫu số (*denominator*) là số nguyên
- Các hàm thành viên (**public**):
  - + Constructor tham số với giá trị ngầm định là tử số = 0 và mẫu số = 1
  - + Copy constructor
  - + Destructor
- Hàm thành viên (**private**):
  - + *reduce()* để đưa số hữu tỉ về dạng tối giản
- Hàm toàn cục:
  - Tìm ước chung lớn nhất (Greatest Common Divisor) để hàm *reduce()* gọi khi tối giản phân số.
  - Hàm này có prototype như sau:  
`int gCD(int, int);`

**Thực hiện quá tải các toán tử sau:**

- Toán tử gán
- Toán tử +, -, \*, / hai số hữu tỉ
- Toán tử so sánh: ==, >, <, !=
- Toán tử cộng một số nguyên với một số hữu tỉ cho kết quả là một số hữu tỉ (đã được rút gọn)
- Toán tử cộng một số hữu tỉ với một số nguyên cho kết quả là một số hữu tỉ (đã được rút gọn)
- Toán tử xuất/nhập số hữu tỉ (<</>>)

Ghi chú:

- Tập tin Rational.h chứa giao tiếp (interface) public cho lớp Rational.
- Tập tin Rational.cpp chứa cài đặt (implementation) của các hàm thành viên, hàm toán tử, hàm bạn, ...
- Tập tin RationalTest.cpp là tập tin của client/uer dùng để test lớp Rational.

Tập tin này có nội dung như sau:

```
#include " Rational.h"
#include <iostream>
using namespace std;
void main()
{
    //Tạo đối tượng
    Rational a(2,5);
    Rational b(8,5);
    Rational c(2,1);
    Rational d(c);
    Rational e;
    Rational i;

    cout << "the first rational (A) should be 2/5 : " << a << endl;
    cout << "the second rational (B) should be 8/5 : " << b << endl;
    cout << "the third rational (C) should be 2 : " << c << endl;
    cout << "the fourth rational (D) should be like (c) = 2 : " << d << endl;
    cout << "the fifth rational (E) should be 0 : " << e << endl;

    cout << "enter I = " << endl;
    cin >> i;
    cout << i << endl;
    cout << " Arithmathics operations " << endl;
    cout << " ===== " << endl;
    cout << a << "+" << b << "=" << a + b << endl;
```

```

cout << c<<"- "<<b<<"=" << c - b << endl;
cout << b<<"- "<<c<<"=" << b - c << endl;
cout << c<<"*"<<b<<"=" << c * b << endl;
cout << b<<"/"<<a<<"=" << b / a << endl;

cout << " Logical operations " << endl;
cout << " ===== " << endl;
cout <<a<<">"<<b<<" should be false: " << (a>b) << endl;
cout << a<<"<"<<b<<" should be True: " << (a<b) << endl;
cout << a<<"=="<<b<<" should be false: " << (a==b) << endl;
cout << a<<"!="<<b<<" should be True: " << (a!=b) << endl;
cout << c<<"=="<<d<<" should be True: " << (c==d) << endl;
cout << " ===== " << endl;

cout << " Arithmathics operations " << endl;
cout << " ===== " << endl;
cout << a<<"+2=" << a + 2 << endl;
cout << " 2 + "<<a<<"=" << 2 + a << endl;
cout << " ===== " << endl;

cout << "\n\nDivision by Zero" << endl;
cout << " ===== " << endl;
cout << "Rational g(3,0): ";
Rational g(3,0);
cout << g << endl;
Rational h(0,1);
cout << "Divide by zero - a/h should not give abnormal exit:" << a/h << endl;
cout << " ===== " << endl;
cout << "Good Luck" << endl;

}

```