

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN



BÁO CÁO THỰC HÀNH 2 LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

Sinh viên: Đồng Quốc Thắng - 23521421

Giáo viên hướng dẫn: Nguyễn Ngọc Quý

Ngày ... tháng ... năm ...



Mục lục

1 Bài 1	3
1.1 Class Diagram	3
1.2 Input, Output và Solution	4
1.3 Code	4
1.4 TestCase	4
2 Bài 2	5
2.1 Class Diagram	5
2.2 Input, Output và Solution	5
2.3 Code	6
2.4 TestCase	6
3 Bài 3	6
3.1 Class Diagram	7
3.2 Input, Output và Solution	7
3.3 Code	7
3.4 TestCase	8
4 Bài 4	8
4.1 Class Diagram	9
4.2 Input, Output và Solution	9
4.3 Code	9
4.4 TestCase	10



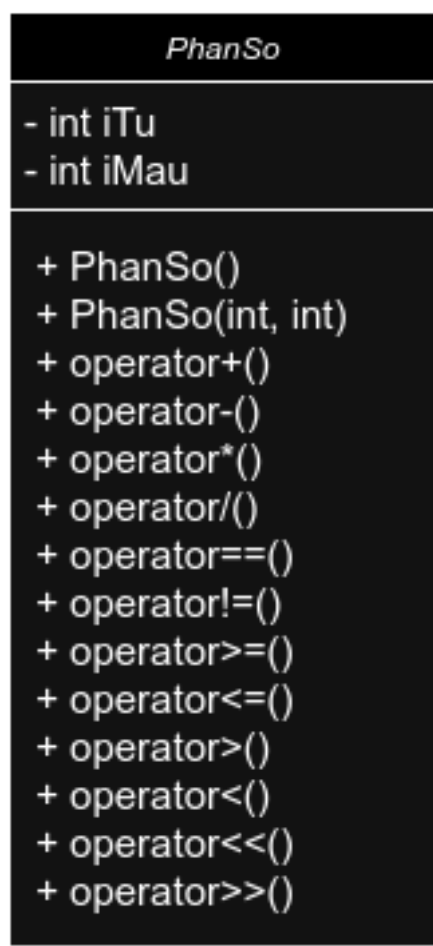
1 Bài 1

Xây dựng lớp phân số:

- Thuộc tính: iTu, iMau
- Phương thức: PhanSo(), PhanSo(int Tu, int Mau)
- Thực hiện các phương thức operator: +, -, *, /, ==, !=, >=, <=, >, <, », «

Yêu cầu: Thực hiện xây dựng lớp, vẽ class diagram và khai báo các thuộc tính, phương thức. Viết nội dung vào các phương thức đã khai báo. Gọi các phương thức trong hàm main()

1.1 Class Diagram





1.2 Input, Output và Solution

Input: Tử và mẫu của phân số ở dạng số nguyên

Output: Sau khi thực hiện các phép toán trên phân số, in ra phân số bằng cách in ra tử và mẫu của phân số

Solution: Tạo class phân số với 2 constructor khác nhau, một constructor không có tham số và 1 constructor có tham số để có thể khởi tạo các phân số thay vì nhập xuất nếu muốn. Các operator « và » được define là hàm bạn, vì các operator này cần sử dụng istream và ostream, để istream và ostream nhập và xuất thông tin cho class này được, thì istream và ostream phải truy cập được dữ liệu private của lớp này là phần thực và phần ảo, do đó phải để là hàm bạn.

1.3 Code

https://github.com/LowTechTurtle/IT002_OOP/tree/main/Lab3/bai1

1.4 TestCase

```
(~/IT002_OOP/Lab3/bai1)
(21:42:18 on main *)→ ./main
Nhapphan so thu nhât:
Nhap tu: 2
Nhap mau: 0
Mau so khong the bang 0, hay nhap lai: 3
Nhapphan so thu hai:
Nhap tu: 2
Nhap mau: -1
tong cua 2phan so: -4/3
hieucua 2phan so: 8/3
tich cua 2phan so: -4/3
Thuong cua 2phan so: -1/3
2phan so khong bang nhau
(~/IT002_OOP/Lab3/bai1)
(21:42:28 on main *)→ ./main
Nhapphan so thu nhât:
Nhap tu: 0
Nhap mau: 0
Mau so khong the bang 0, hay nhap lai: 4
Nhapphan so thu hai:
Nhap tu: -9
Nhap mau: 2
tong cua 2phan so: -9/2
hieucua 2phan so: 9/2
tich cua 2phan so: 0
Thuong cua 2phan so: 0
2phan so khong bang nhau
(~/IT002_OOP/Lab3/bai1)
(21:42:36 on main *)→
```



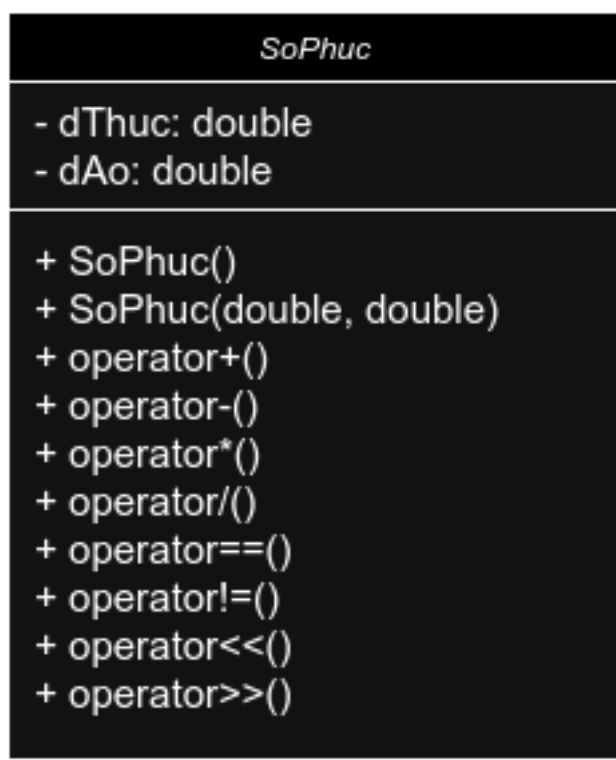
2 Bài 2

Xây dựng lớp số phức:

- Thuộc tính: dThuc, dAo
- Phương thức: SoPhuc(), SoPhuc(int thuc, int ao)
- Thực hiện các phương thức operator: +, -, *, /, ==, !=, », «

Yêu cầu: Thực hiện xây dựng lớp, vẽ class diagram và khai báo các thuộc tính, phương thức. Viết nội dung vào các phương thức đã khai báo. Gọi các phương thức trong hàm main()

2.1 Class Diagram



2.2 Input, Output và Solution

Input: Nhập vào phần thực, phần ảo của số phức

Output: Sau khi gọi các phương thức trên các số phức, in các số phức ra màn hình

Solution: Tạo một class cho số phức, tạo 2 biến private là dThuc và dAo, sau đó xây dựng các constructor để có thể khởi tạo phần thực và ảo. Các operator « và » được define là hàm bạn, lý do là hàm bạn giống như ở bài 1.



2.3 Code

https://github.com/LowTechTurtle/IT002_OOP/tree/main/Lab3/bai2

2.4 TestCase

```
(21:50:15 on main *)→ ./main
Nhap so thu nhat:
Nhap phan thuc: -3
Nhap phan ao: 2
Nhap so thu hai:
Nhap phan thuc: 1
Nhap phan ao: 4
So thu nhat:
-3 + 2i
So thu hai:
1 + 4i
Tong: -2 + 6i
Hieu: -4 - 2i
Tich: -11 - 10i
Thuong: 0.294118 + 0.823529i
Hai so khong bang nhau
~/IT002_OOP/Lab3/bai2
(21:50:22 on main *)→ ./main
Nhap so thu nhat:
Nhap phan thuc: 15
Nhap phan ao: 48
Nhap so thu hai:
Nhap phan thuc: -12
Nhap phan ao: 0
So thu nhat:
15 + 48i
So thu hai:
-12 + 0i
Tong: 3 + 48i
Hieu: 27 + 48i
Tich: -180 - 576i
Thuong: -1.25 - 4i
Hai so khong bang nhau
```

3 Bài 3

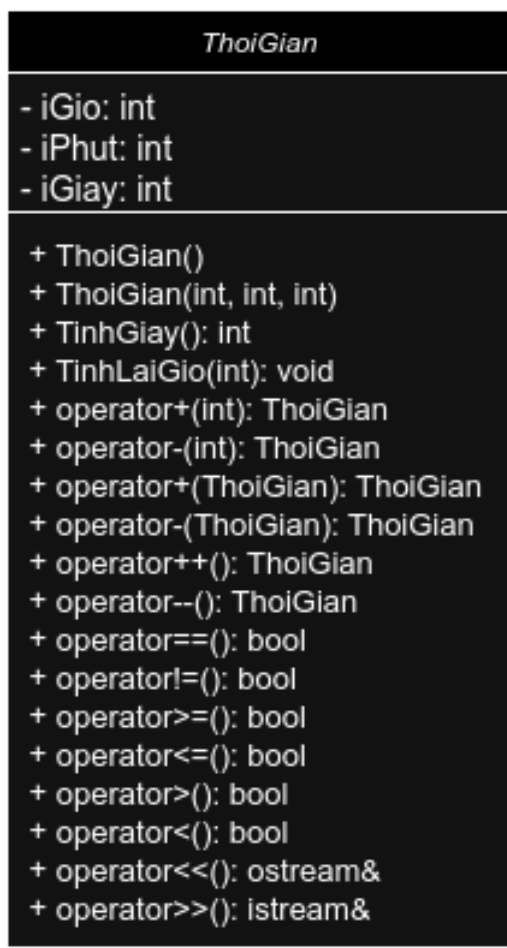
Xây dựng lớp thời gian:

- Thuộc tính: iGio, iPhut, iGiay
- Phương thức: ThoiGian(), ThoiGian (int Gio, int Phut, int Giay), TinhGiay(), TinhLaiGio(int Giay)
- Thực hiện các phương thức operator: +(int Giay), -(int Giay), +(ThoiGian a), -(ThoiGian a), ++, --, ==, !=, >=, <=, >, <, », «



Yêu cầu: Thực hiện xây dựng lớp, vẽ class diagram và khai báo các thuộc tính, phương thức. Viết nội dung vào các phương thức đã khai báo. Gọi các phương thức trong hàm main()

3.1 Class Diagram



3.2 Input, Output và Solution

Input: Giờ phút giây của các thời điểm

Output: Giờ phút giây của các instances sau khi đã thực hiện các phép toán

Solution: Sử dụng số giây trong ngày để so sánh các thời điểm trong ngày. Số giây nào nhỏ hơn thì thời điểm đó đến trước trong ngày. Sử dụng 24h format. Các hàm như tính lại giờ hay cộng trừ thêm giây, đều được đưa về số giây đã qua trong ngày và convert lại thành giờ phút giây.

3.3 Code

https://github.com/LowTechTurtle/IT002_00P/tree/main/Lab3/bai3



3.4 TestCase

```
(~/IT002_OOP/Lab3/bai3) —————  
[ (22:02:51 on main ★) —> ./main  
Nhập thời điểm đầu tiên  
Nhập giờ: 5  
Nhập phút: -31  
Phút phải từ 0 đến 59, hãy nhập lại  
31  
Nhập giây: 444  
Giây phải từ 0 đến 59, hãy nhập lại  
44  
Nhập thời điểm thứ hai  
Nhập giờ: 2  
Nhập phút: 60  
Phút phải từ 0 đến 59, hãy nhập lại  
6  
Nhập giây: 1  
Thời điểm đầu tiên muộn hơn  
Thời điểm đầu tiên sau khi tăng 1 giây:  
Giờ: 5  
Phút: 31  
Giây: 45  
Thời điểm thứ hai sau khi giảm 1 giây:  
Giờ: 2  
Phút: 6  
Giây: 0
```

4 Bài 4

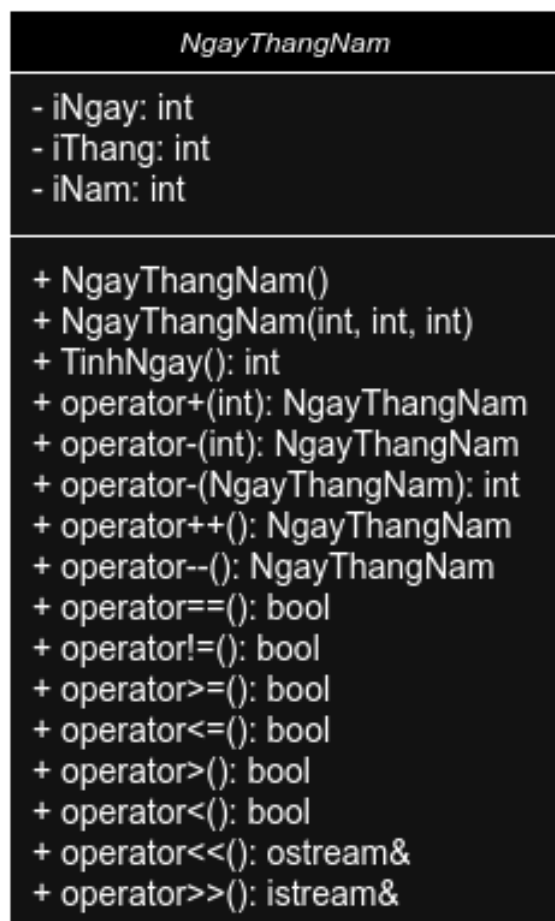
Xây dựng lớp ngày tháng năm:

- Thuộc tính: iNgày, iThang, iNam
- Phương thức: NgayThangNam(), NgayThangNam (int Nam, int Thang, int Ngay), TinhNgay()
- Thực hiện các phương thức operator: +(int ngay), -(int ngay), -(NgayThangNam a), ++, --, ==, !=, >=, <=, >, <, », «



Yêu cầu: Thực hiện xây dựng lớp, vẽ class diagram và khai báo các thuộc tính, phương thức. Viết nội dung vào các phương thức đã khai báo. Gọi các phương thức trong hàm main()

4.1 Class Diagram



4.2 Input, Output và Solution

Input: Ngày tháng năm

Output: Ngày tháng năm của các instances sau khi đã thực hiện các phép toán

Solution: Sử dụng số ngày từ năm 0 để so sánh các thời điểm. Số ngày nào nhỏ hơn thì thời điểm đó đến trước. Sử dụng switch case để tìm số ngày có trong các tháng của năm. Khi cộng hay trừ bớt số ngày, xét các trường hợp tăng/giảm số tháng khi đã quá ngày trong tháng. Nếu tăng tháng thì phải để ý tháng có phải là tháng 12 không để tăng năm. Khi giảm ngày thì phải để ý tháng có phải tháng 1 không để giảm năm.

4.3 Code

https://github.com/LowTechTurtle/IT002_00P/tree/main/Lab3/bai4



4.4 TestCase

Log của output thứ nhất(tc 1):

```
1  Nhap ngay thu nhat:
2  Nhap nam: 2024
3  Nhap thang: 2
4  Nhap ngay: 100
5  Ngay trong thang khong hop le, hay nhap lai
6  5
7  Nhap ngay thu hai:
8  Nhap nam: -2
9  Nam khong the am, hay nhap lai:
10 2023
11 Nhap thang: 11
12 Nhap ngay: 1
13 Ngay1: Ngay: 5
14 Thang: 2
15 Nam: 2024
16
17 Ngay2: Ngay: 1
18 Thang: 11
19 Nam: 2023
20
21 Ngay thu hai den truoc
22 Ngay thu nhat sau 100 ngay: Ngay: 16
23 Thang: 5
24 Nam: 2024
25
26 100 ngay truoc ngay thu hai: Ngay: 24
27 Thang: 7
28 Nam: 2023
29
30 Ngay sau cua ngay thu nhat: Ngay: 17
31 Thang: 5
32 Nam: 2024
33
34 Ngay truoc cua ngay thu hai: Ngay: 23
35 Thang: 7
36 Nam: 2023
```

Log của output thứ 2(tc 2)

```
1  Nhap ngay thu nhat:
2  Nhap nam: 2023
3  Nhap thang: 11
4  Nhap ngay: 22
5  Nhap ngay thu hai:
6  Nhap nam: 2024
```



⁷Nhap thang: 2
⁸Nhap ngay: 18
⁹Ngay1: Ngay: 22
¹⁰Thang: 11
¹¹Nam: 2023
¹²
¹³Ngay2: Ngay: 18
¹⁴Thang: 2
¹⁵Nam: 2024
¹⁶
¹⁷Ngay thu nhat den truoc
¹⁸Ngay thu nhat sau 100 ngay: Ngay: 2
¹⁹Thang: 3
²⁰Nam: 2024
²¹
²²100 ngay truoc ngay thu hai: Ngay: 10
²³Thang: 11
²⁴Nam: 2023
²⁵
²⁶Ngay sau cua ngay thu nhat: Ngay: 3
²⁷Thang: 3
²⁸Nam: 2024
²⁹
³⁰Ngay truoc cua ngay thu hai: Ngay: 9
³¹Thang: 11
³²Nam: 2023
