

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



BÁO CÁO THỰC HÀNH 4

LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

Sinh viên: Đồng Quốc Thắng - 23521421

Giáo viên hướng dẫn: Nguyễn Ngọc Quý



Mục lục

1	Bài 1	3
1.1	Class diagram	3
1.2	Input, Output, Solution	3
1.3	Code	4
1.4	Testcase	4
2	Bài 2	5
2.1	Class diagram	5
2.2	Input, Output, Solution	5
2.3	Code	5
2.4	Testcase	6
3	Bài 3	6
3.1	Class diagram	7
3.2	Input, Output, Solution	7
3.3	Code	7
3.4	Testcase	8



1 Bài 1

Quản lý thông tin nhân viên

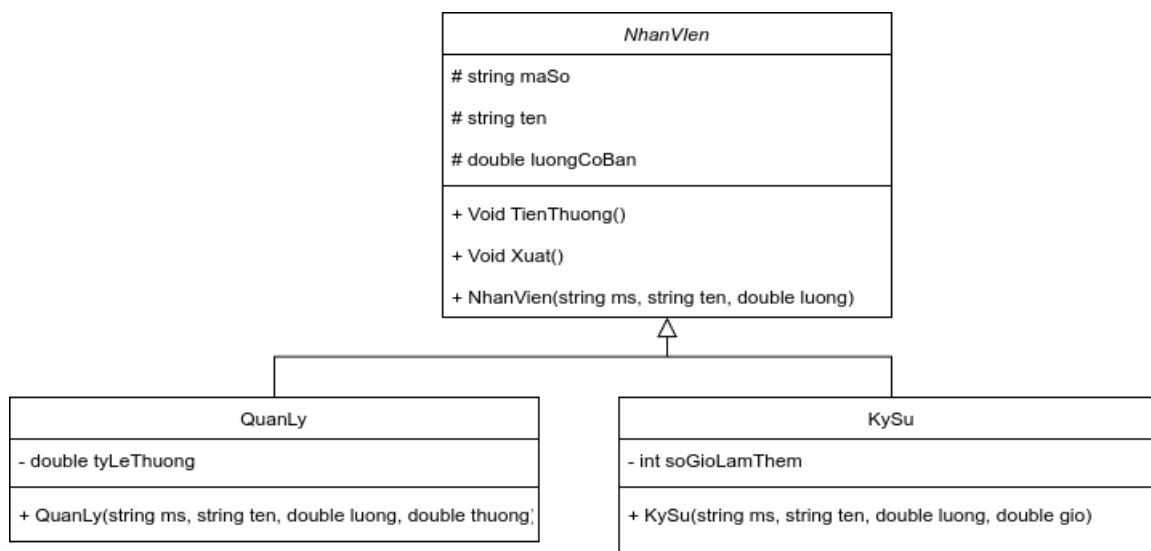
Viết một chương trình C++ để quản lý thông tin nhân viên trong một công ty. Sử dụng tính kế thừa để tạo lớp cơ sở `NhanVien` và hai lớp con `Quản lý` và `Kỹ sư`. Mỗi loại nhân viên có các thuộc tính như sau:

- Nhân viên: Mã số nhân viên, Tên, Lương cơ bản.
- Quản lý: Thừa kế từ Nhân viên với thêm thuộc tính là Tỷ lệ thưởng. Phương thức `TienThuong()` tính toán tiền thưởng dựa trên tỷ lệ thưởng và lương cơ bản.
- Kỹ sư: Thừa kế từ Nhân viên với thêm thuộc tính là Số giờ làm thêm. Phương thức `TienThuong()` tính toán tiền thưởng dựa trên số giờ làm thêm (mỗi giờ làm thêm được trả 100.000).

Chương trình cần thực hiện các công việc sau:

- Cho phép người dùng nhập thông tin của một quản lý và một kỹ sư.
- Hiển thị thông tin chi tiết của cả hai nhân viên, bao gồm mã số, tên, lương cơ bản và tiền thưởng tính được.
- Sử dụng tính kế thừa để thực hiện việc tính toán tiền thưởng dựa trên loại nhân viên.
- Sử dụng hàm `Xuat()` để hiển thị thông tin của nhân viên.

1.1 Class diagram



1.2 Input, Output, Solution

Input: Nhập thông tin của một quản lý và một kỹ sư. Thông tin của quản lý gồm : Mã số nhân viên, Tên, Lương cơ bản, Tỷ lệ thưởng. Kỹ sư gồm : Mã số nhân viên, Tên, Lương cơ bản, Số giờ làm thêm.

Output: Hiển thị thông tin chi tiết của cả hai nhân viên, bao gồm mã số, tên, lương cơ bản và tiền thưởng tính được.



Solution: Xây dựng lớp nhân viên với 3 thuộc tính là mã số, tên, lương cơ bản. Với các hàm virtual là tiền thưởng và xuất thông tin. Sau đó cho các lớp Quản lý và kỹ sư kế thừa lớp nhân viên. Quản lý có thuộc tính mới là tỷ lệ thưởng và Kỹ sư có thuộc tính mới là số giờ làm thêm

1.3 Code

https://github.com/LowTechTurtle/IT002_OOP/tree/main/Lab4/bai1

1.4 Testcase

```
(~/IT002_OOP/Lab4/bai1)
(16:58:29 on main)→ ./main
Nhap thong tin quan ly:
Ma so: 123
Ten: Nguyen Van A
Luong co ban: 5000000
Ty le thuong: 1.25

Nhap thong tin ky su:
Ma so: 213
Ten: Nguyen Van B
Luong co ban: 6000000
So gio lam them: 20

Thong tin quan ly:
Ma so: 123
Ten: Nguyen Van A
Luong co ban: 5e+06
Ty le thuong: 1.25
Tien thuong: 6.25e+06

Thong tin ky su:
Ma so: 213
Ten: Nguyen Van B
Luong co ban: 6e+06
So gio lam them: 20
Tien thuong: 2e+06
(~/IT002_OOP/Lab4/bai1)
(16:59:07 on main)→ |
```



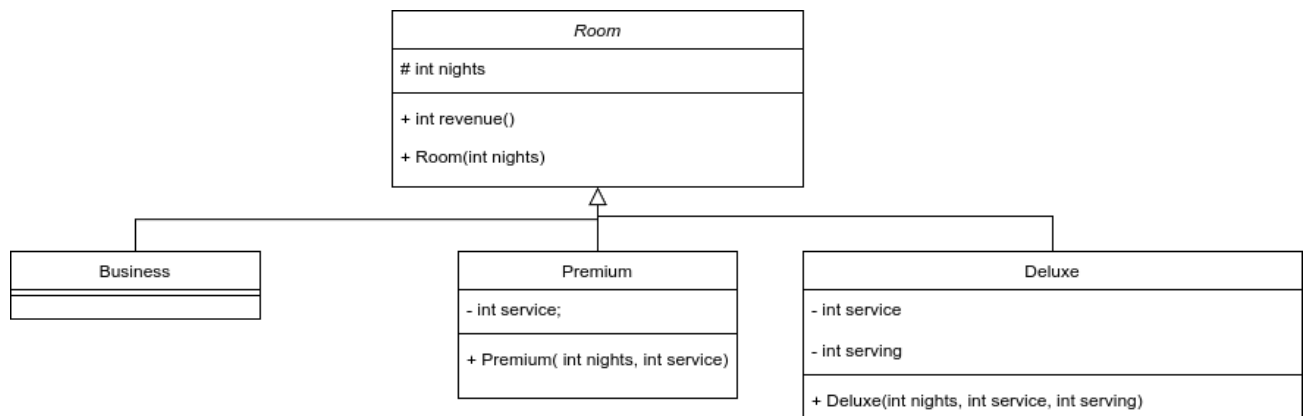
2 Bài 2

Khách sạn SOfITEL có 3 loại phòng với doanh thu phòng của tháng được tính như sau:

- Loại Deluxe: doanh thu phòng = Số đêm * 750000 + Phí dịch vụ + Phí phục vụ
- Loại Premium: doanh thu phòng = Số đêm * 500000 + Phí dịch vụ
- Loại Business: doanh thu phòng = Số đêm * 300000

Yêu cầu: Hàm main: khởi tạo 5 phòng: Ví dụ: Deluxe a(...); Deluxe b(...); Premium c(...); Premium d(...); Business e(...). Hãy cho biết doanh thu của loại phòng nào nhiều nhất? (lưu ý: mỗi loại phòng có thể khởi tạo nhiều phòng).

2.1 Class diagram



2.2 Input, Output, Solution

Input: Bài này không yêu cầu nhập liệu mà viết sẵn và khởi tạo 5 phòng với 2 phòng 2 deluxe, 2 phòng premium và 1 phòng business

Output: Hiển thị ra loại phòng có doanh thu cao nhất và cho biết doanh thu đó là bao nhiêu.

Solution: Tạo một lớp là Room với 1 thuộc tính là số đêm ở, 1 hàm virtual là doanh thu. Sau đó cho các lớp phòng kế thừa lớp Room, phòng Premium có thêm thuộc tính service, phòng Deluxe có thêm thuộc tính service và serving.

2.3 Code

https://github.com/LowTechTurtle/IT002_OOP/tree/main/Lab4/bai2



2.4 Testcase

```
turtle@EvilBrewHausen:~/IT002
[~/IT002_OOP/Lab4/ba12]—
[ (16:59:59 on main)→ ./main
Loại phòng có doanh thu cao nhất là Deluxe với doanh thu là 6420000 VNĐ.
[~/IT002_OOP/Lab4/ba12]—
[ (17:00:00 on main)→ |
```

3 Bài 3

Một nông trại chăn nuôi có 3 loại gia súc: bò, cừu, và dê. Mỗi loại gia súc đều có thể sinh con, cho sữa và phát ra tiếng kêu riêng của chúng. Khi đói, các gia súc sẽ phát ra tiếng kêu để đòi ăn. Sau một thời gian chăn nuôi, người chủ nông trại muốn thống kê xem trong nông trại có bao nhiêu gia súc ở mỗi loại, tổng số sữa mà tất cả các gia súc của ông đã cho. Áp dụng kế thừa, xây dựng chương trình cho phép người chủ nông trại nhập vào số lượng gia súc ban đầu ở mỗi loại.

a) Một hôm người chủ nông trại đi vắng, tất cả gia súc trong nông trại đều đói. Hãy cho biết những tiếng kêu nghe được trong nông trại.

b) Chương trình sẽ đưa ra thống kê các thông tin người chủ mong muốn (nêu trên) sau một lứa sinh và một lượt cho sữa của tất cả gia súc. Biết rằng:

- Tất cả gia súc ở mỗi loại đều sinh con.
- Số lượng sinh của mỗi gia súc là ngẫu nhiên.
- Tất cả gia súc mỗi loại đều cho sữa.
- Số lượng sinh của mỗi gia súc cho sữa là ngẫu nhiên nhưng trong giới hạn sau

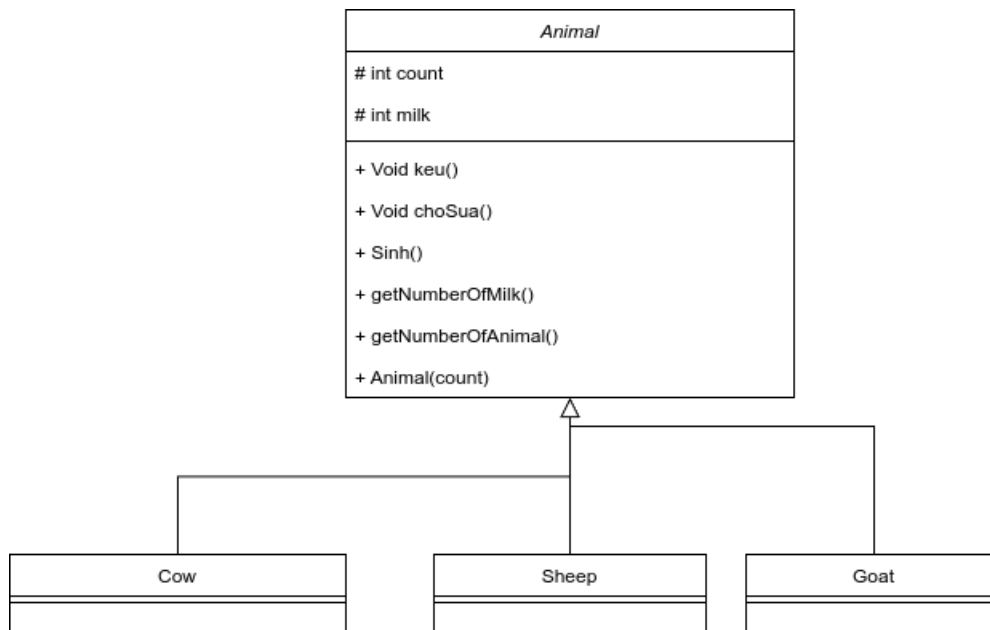
Bò: 0 - 20 lít

Cừu: 0 - 5 lít

Dê: 0 - 10 lít



3.1 Class diagram



3.2 Input, Output, Solution

Input: Nhập vào số lượng bò, cừu, dê.

Output: Số bò, cừu, dê và số lượng sữa mỗi loại gia súc cho được sau một lứa sinh và một lượt cho sữa.

Solution: Tạo 1 lớp **Animal** với các thuộc tính `count`, `milk`. Các hàm virtual như `kêu`, `cho sữa`, `sinh`. Sau đó cho các lớp con là bò, dê, cừu kế thừa lớp **Animal**.

3.3 Code

https://github.com/LowTechTurtle/IT002_00P/tree/main/Lab4/bai3



3.4 Testcase

```
(~/IT002_OOP/Lab4/bai3)
(17:00:43 on main)→ ./main
Nhap so con bo: 3
Nhap so con cuu: 5
Nhap so con de: 6
Bo doi
Bo doi
Bo doi
Cuu doi
Cuu doi
Cuu doi
Cuu doi
Cuu doi
De doi
De doi
De doi
De doi
De doi
De doi
So con bo la: 5
So sua con bo cho la: 56
So con cuu la: 11
So sua con cuu cho la: 31
So con de la: 15
So sua con de cho la: 95
(~/IT002_OOP/Lab4/bai3)
(17:00:47 on main)→ |
```