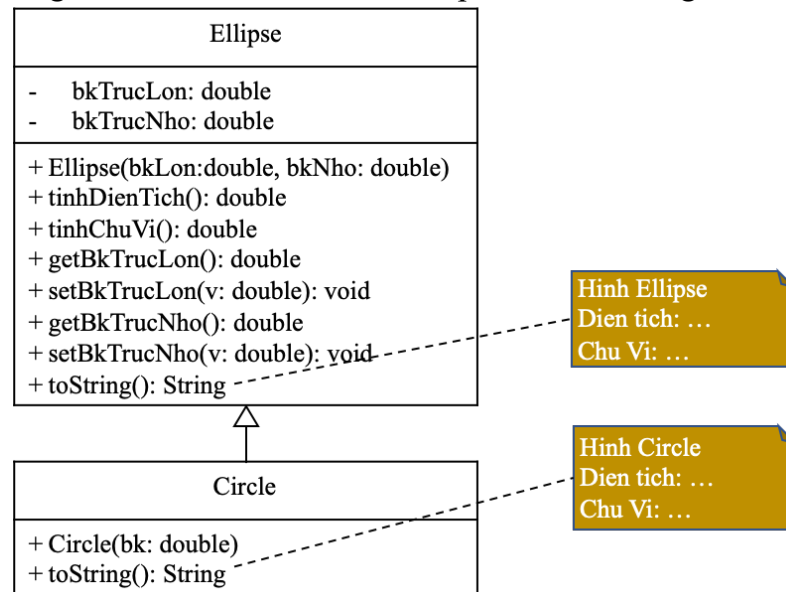


## Bài thực hành 3 – Kế thừa & đa hình

**Bài 1:** Viết chương trình hiện thực hoá sơ đồ lớp bên dưới, trong đó



Lớp `Ellipse` có 2 thuộc tính bán kính trục lớn, bán kính trục bé và các phương thức

- Phương thức khởi tạo 2 tham số khởi gán cho bán kính hai trục.
- Phương thức tính diện tích và chu vi của hình Ellipse.
- Các phương thức getter và setter của các thuộc tính.
- Ghi đè phương thức `toString()` trả về chuỗi đại diện cho một Ellipse có định dạng như phần ghi chú trong hình.

Biết công thức tính chu vi và diện tích của ellipse như sau:

- Diện tích =  $PI * bkTrucLon * bkTrucNho$
- Chu vi =  $2 * PI * \sqrt{\frac{bkTrucLon^2 + bkTrucNho^2}{2}}$

Lớp `Circle` kế thừa lớp `Ellipse` có các phương thức

- Phương thức khởi tạo 1 tham số để khởi gán bán kính hình tròn.
- Ghi đè phương thức `toString()` trả về chuỗi đại diện cho một Circle như trong phần ghi chú.

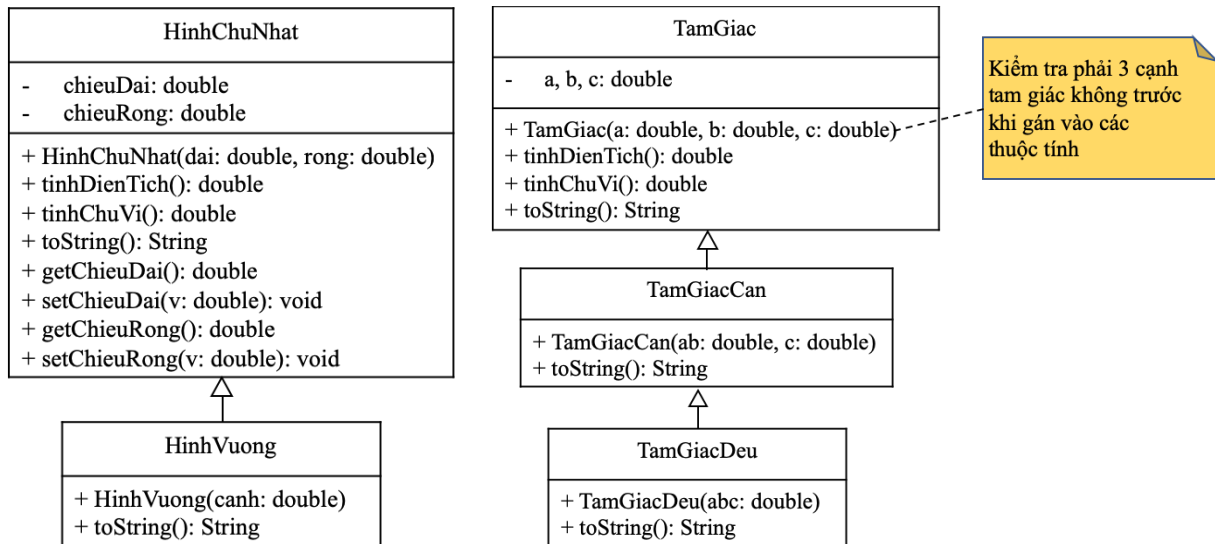
Viết phương thức `main()` kiểm tra các chức năng trong các lớp.

**Bài 2:** Tương tự bài 1, hiện thực hoá 2 sơ đồ lớp bên dưới.

Biết công thức tính diện tích của tam giác khi biết ba cạnh  $a, b, c$ .

$$S = \sqrt{p(p-a)(p-b)(p-c)}$$

Trong đó  $p = \frac{a+b+c}{2}$  là nửa chu vi.



**Bài 3:** Một công ty chuyên sản xuất sách và băng đĩa phục vụ học tập trong nhà trường phổ thông. Mỗi sản phẩm đều có mã sản phẩm, tên sản phẩm, mô tả sản phẩm, nhà sản xuất, giá bán. Ngoài ra:

- Sách có thêm thông tin số trang.
- Băng đĩa có thêm thông tin độ dài thời gian phát.

Xây dựng lớp cho phép quản lý danh sách sản phẩm tại công ty có các chức năng.

- Nhập thêm sản phẩm vào danh sách.
- Xóa sản phẩm khỏi danh sách.
- Xuất thông tin sản phẩm. Mỗi sản phẩm yêu cầu hiển thị đầy đủ thông tin từng thuộc tính của sản phẩm đó, mỗi thuộc tính hiển thị trên một dòng.
- Tìm kiếm thông tin sản phẩm thông qua tên, mô tả, ngày phát hành sản phẩm, loại sản phẩm (sách hoặc băng đĩa).
- Sắp xếp danh sách sản phẩm giảm dần theo giá.

Viết phương thức `main()` chứa menu sau để người dùng chọn chức năng thực hiện.

1. Thêm sản phẩm.
2. Xóa sản phẩm, các phương thức cho phép truyền vào đối tượng sản phẩm hoặc chỉ là mã sản phẩm để xóa.
3. Cập nhật sản phẩm tên sản phẩm, mô tả sản phẩm hoặc giá sản phẩm dựa trên mã sản phẩm.
4. Tìm kiếm sản phẩm theo mã sản phẩm, tên sản phẩm hoặc khoảng giá bán.
5. Sắp xếp các sản phẩm giảm dần theo giá.
6. Xem danh sách sản phẩm.
7. Thoát.

Trong đó, chức năng thêm sách người dùng sẽ nhập thông tin sách cần thêm, xóa sách người dùng nhập mã sản phẩm, tìm kiếm sách người dùng nhập từ khóa muốn tìm.

**Bài 4:** Một hệ thống ngân hàng gồm 2 loại tài khoản chính là tài khoản có kỳ hạn và tài khoản không kỳ hạn. Một tài khoản bao gồm các thông tin số tài khoản, tên tài khoản,

điện thoại, email, số tiền tài khoản, ngày tạo tài khoản, trạng thái tài khoản. Riêng tài khoản có kỳ hạn có thêm thông tin kỳ hạn và ngày đáo hạn.

Trong đó số tài khoản là chuỗi có 6 số tính từ 00001 và tăng dần khi tạo tài khoản mới. Thông tin kỳ hạn cho phép được lấy từ enum sau:

```
enum KyHan {  
    MOT_TUAN,  
    MOT_THANG,  
    MOT_NAM  
}
```

Viết chương trình thực hiện các chức năng:

- Xem thông tin tài khoản gồm thông tin số tài khoản, tên tài khoản, số tiền và loại tài khoản, nếu là tài khoản có kỳ hạn thì hiển thị thêm thông tin kỳ hạn và ngày đáo hạn.
- Tra cứu tài khoản theo số tài khoản và tên tài khoản.
- Nộp tiền vào tài khoản được nạp ngay đối với tài khoản không kỳ hạn, nhưng tài khoản có kỳ hạn chỉ được nạp vào trong ngày đáo hạn.
- Rút tiền thì được rút ngay đối với tài khoản không kỳ hạn, nhưng tài khoản có kỳ hạn chỉ được rút trong ngày đáo hạn, nếu người dùng vẫn rút trong ngày đáo hạn thì tài khoản đó sẽ bị tắt toán (trở thành tài khoản không kỳ hạn) và không có tiền lãi trong kỳ hạn hiện tại.
- Tính tiền lãi khi đến ngày đáo hạn, tiền lãi được cộng trực tiếp số tiền trong tài khoản.

Loại tài khoản		Lãi suất
Không kỳ hạn		0.1%/năm
Có kỳ hạn	1 tuần	0.5%/năm
	1 tháng	4.5%/năm
	1 năm	6.8%/năm

- Cập nhật ngày đáo hạn đối với tài khoản có kỳ hạn.

**Bài 5:** Viết chương trình quản lý thông tin tiền lương nhân viên làm việc thời vụ của công ty XYZ, biết một nhân viên bao gồm các thông tin số mã số, chứng minh nhân dân, họ tên, quê quán, giới tính, số điện thoại liên lạc. Trong đó mã số là số nguyên tự động tăng mỗi khi thêm nhân viên mới.

Nhân viên bình thường được tính lương bằng  $\langle \text{lương} \rangle * \langle \text{số ngày công} \rangle$

Tuy nhiên nhân viên thuộc một trong 3 bộ phận A, B hoặc C được tính như sau:

- Bộ phận A:  $1.2 * \langle \text{lương} \rangle * \langle \text{số ngày công} \rangle$
- Bộ phận B:  $1.5 * \langle \text{lương} \rangle * \langle \text{số ngày công} \rangle$
- Bộ phận C:  $2.5 * \langle \text{lương} \rangle * \langle \text{số ngày công} \rangle$

Trong đó  $\langle \text{lương} \rangle$  là mức lương chuẩn cho mỗi ngày làm, quy định công ty là 100.000 VNĐ/ngày và nhân viên sẽ được điểm danh mỗi ngày.

Chương trình bao gồm các chức năng sau:

- Thêm và xóa nhân viên mới của một bộ phận trong danh sách nhân viên.
- Hiển thị danh sách nhân viên, mỗi nhân viên hiển thị các thông tin: họ tên, số chứng minh nhân dân, giới tính, quê quán, số điện thoại, bộ phận.

- Tra cứu nhân viên theo số chứng minh nhân dân, họ tên hoặc quê quán.
- Tính tiền lương cho các nhân viên. Trong phương thức này, người dùng sẽ nhập số ngày công của mỗi nhân viên và hiển thị mã số, họ tên, số ngày công và tiền lương của từng nhân viên.

**Chú ý:** *Không dùng if/else và switch/case tính tiền lương.*

Viết phương thức `main()` chứa menu sau để người dùng chọn chức năng thực hiện

1. Thêm nhân viên.
2. Xoá nhân viên.
3. Tra cứu nhân viên.
4. Hiển thị danh sách nhân viên.
5. Tính lương nhân viên.
6. Thoát.

Sau một thời gian phát triển chương trình, công ty XYZ yêu cầu mở thêm bộ phận D với cách tính lương như sau:  $\text{<lương> * <số ngày làm> + <phụ cấp>}$ , tùy vào năng lực của mỗi nhân viên lúc phỏng vấn sẽ có mức lương phụ cấp khác nhau.

Hãy cập nhật chương trình đã viết nhưng hạn chế tối đa sửa mã nguồn để thêm bộ phận D, cũng như quản lý tiền lương cho các nhân viên thuộc bộ phận D.