*Sinh viên có thể tự thêm các hàm, tự chọn danh sách tham số, tự đặt chế độ private protected public hợp lý để đảm bảo tính hướng đối tượng (chỉ trừ các thành viên đã nêu rõ private hay protected thì cần giữ nguyên)*

Cho lớp trừu tượng Animal, gồm các thành viên sau:

* Các thuộc tính private: cân nặng xuất chuồng, cân nặng hiện tại
* Hàm setInfor thiết lập giá trị cho mọi thuộc tính
* Hàm ảo thuần tuý Feed, thực hiện hành động cho Animal ăn, truyền vào tham số là loại thức ăn

Lớp Chicken kế thừa Animal, có:

* Thuộc tính private: giống gà
* Hàm setInfor thiết lập giá trị cho mọi thuộc tính
* Override hàm Feed, biết gà chỉ ăn thóc:
* nếu thực phẩm khác thóc thì in ra cảnh báo “Gà không ăn <giá\_trị\_truyền\_vào>”
* nếu thực phẩm là thóc, thì in ra “Gà đã ăn no”, tăng cân nặng hiện tại của gà lên 1, nếu cân nặng >= mức xuất chuồng thì in ra “Cần cho gà xuất chuồng”

Lớp Cow kế thừa Animal, có:

* Thuộc tính private: hệ số cân nặng
* Hàm setInfor thiết lập giá trị cho mọi thuộc tính
* Override hàm Feed, biết bò chỉ ăn cỏ:
* nếu thực phẩm khác cỏ thì in ra cảnh báo “Bò không ăn <giá\_trị\_truyền\_vào>”
* nếu thực phẩm là cỏ, thì in ra “Bò đã ăn no”, tăng cân nặng hiện tại của bò lên bằng hệ số cân nặng, nếu cân nặng hiện tại lúc này >= mức xuất chuồng thì in ra “Cần cho bò xuất chuồng”

Hàm main sẽ tạo một đàn 10 Animal có cả Chicken và Cow, sau cùng cho cả đàn ăn 3 lần.