

Bài tập chương 2

I) Tạo kiểu lớp theo mô tả

Bài 1.

(Tài khoản) Tạo lớp có tên Account đại diện cho tài khoản ngân hàng của khách hàng. Lớp bao gồm một dữ liệu thành phần kiểu int để thể hiện số dư tài khoản. Cung cấp một hàm tạo có tham số là số dư ban đầu và sử dụng nó để khởi tạo dữ liệu thành phần. Hàm tạo cần kiểm tra số dư ban đầu để đảm bảo rằng nó lớn hơn hoặc bằng 0. Nếu không, hãy đặt số dư tài khoản là 0 và hiển thị thông báo lỗi cho biết số dư ban đầu không hợp lệ. Cung cấp ba hàm thành phần như sau.

- Hàm credit (ghi có) để thêm một số tiền vào số dư hiện tại.
- Hàm debit (ghi nợ) để rút tiền từ Tài khoản và đảm bảo rằng số tiền ghi nợ không vượt quá số dư hiện tại trong Tài khoản. Nếu trường hợp đó xảy ra, số dư sẽ không thay đổi và hàm sẽ in một thông báo cho biết "Số tiền ghi nợ vượt quá số dư tài khoản".
- Hàm thành phần getBalance để trả về số dư hiện tại.

Trong chương trình chính tạo hai đối tượng kiểu lớp Account và gọi các hàm thành phần của lớp Account để kiểm tra hoạt động của các hàm đó.

Bài 2

(Hóa đơn) Tạo một lớp có tên Invoice mà một cửa hàng máy tính có thể sử dụng để thể hiện hóa đơn cho một mặt hàng được bán tại cửa hàng. Hóa đơn phải bao gồm bốn dữ liệu thành phần: mã mặt hàng (kiểu string), mô tả mặt hàng (string), số lượng mặt hàng được mua (kiểu int) và giá mỗi mặt hàng (kiểu int).

Lớp Invoice có một hàm tạo khởi tạo bốn dữ liệu thành phần trên. Cung cấp một hàm set và một hàm get cho mỗi dữ liệu thành phần. Ngoài ra, cung cấp một hàm thành phần có tên getInvoiceAmount để tính toán số tiền trên hóa đơn (tức là nhân số lượng với giá trên mỗi mặt hàng), sau đó trả về số tiền dưới dạng giá trị int. Nếu số lượng không dương, thì nên đặt thành 0. Nếu giá của mỗi mặt hàng không dương, thì nên đặt thành 0. Viết chương trình chính kiểm tra các hàm của lớp Invoice.

Bài 3

(Tính nhịp tim mục tiêu) Trong khi tập thể dục, bạn có thể sử dụng máy đo nhịp tim để thấy rằng nhịp tim của bạn nằm trong phạm vi an toàn được đề xuất bởi huấn luyện viên và bác sĩ của bạn. Theo Hiệp hội Tim mạch Hoa Kỳ (AHA), công thức tính nhịp tim tối đa của bạn theo số nhịp đập mỗi phút là 220 trừ đi số tuổi của bạn.

Nhịp tim mục tiêu của bạn là một khoảng từ 50 đến 85% nhịp tim tối đa của bạn. [Lưu ý: Những công thức này là ước tính được cung cấp bởi AHA. Nhịp tim tối đa và mục tiêu có thể thay đổi dựa trên sức khỏe, thể lực và giới tính của từng người]

Tạo một lớp tên là `HeartRates`. Các thuộc tính lớp bao gồm tên, họ và ngày sinh của người dùng (bao gồm các thuộc tính riêng biệt cho tháng, ngày và năm sinh). Lớp của bạn nên có một hàm tạo nhận các dữ liệu này dưới dạng tham số. Đối với mỗi thuộc tính cung cấp hàm set và hàm get.

Lớp cũng nên bao gồm một hàm **`getAge`** để tính toán và trả về tuổi của người (tính theo năm), một hàm **`getMaximumHeartRate`** để tính toán và trả về nhịp tim tối đa của người đó và một hàm **`getTargetMinHeartRate`** và **`getTargetMaxHeartRate`** tính toán và trả về khoảng nhịp tim mục tiêu của người đó. Vì bạn chưa biết cách lấy ngày hiện tại từ máy tính, nên hàm `getAge` sẽ nhắc người dùng nhập tháng, ngày và năm hiện tại trước khi tính tuổi.

Viết một chương trình chính nhập thông tin của người đó, khởi tạo một đối tượng của `HeartRates` và in thông tin từ đối tượng đó bao gồm tên, họ và ngày sinh của người đó, sau đó tính toán và in tuổi của người đó, nhịp tim tối đa và khoảng nhịp tim mục tiêu.

Bài 4

(Ngày tháng) Thiết kế một lớp tên là `Date` có các dữ liệu thành phần là số nguyên để lưu trữ tháng, ngày và năm.

Lớp `Date` có một hàm tạo mặc định ba tham số cho phép đặt các giá trị ngày/tháng/năm vào trong đối tượng `Date` được tạo ra. Nếu khi tạo một đối tượng `Date` mà không chuyển bất kỳ tham số nào hoặc nếu bất kỳ giá trị của tham số được truyền mà không hợp lệ, thì sẽ sử dụng các giá trị mặc định là 1, 1, 2001 (tức là ngày 1 tháng 1 năm 2001). Lớp có ba hàm thành phần để in ngày tháng theo các định dạng sau:

- a) 15/03/13,
- b) ngày 15 tháng 3 năm 2013,
- c) 15 tháng 3 năm 2013

Thể hiện hoạt động của lớp Date bằng cách viết một chương trình main sử dụng nó. Hãy chắc chắn rằng chương trình của bạn chỉ chấp nhận các giá trị hợp lệ cho tháng và ngày. Tháng nằm trong khoảng từ 1 đến 12. Ngày cần nằm trong khoảng từ 1 đến số ngày trong tháng đã chọn. (Ví dụ nếu tháng là 1 thì ngày từ 1..31, nếu tháng là 2 và năm không phải năm nhuận thì ngày từ 1..28. Một năm là năm nhuận nếu số năm chia hết cho 400 hoặc là vừa chia hết cho 4 và vừa không chia hết cho 100)

Bài 5

Viết một lớp tên là Car có các biến dữ liệu thành phần sau:

- year Một biến kiểu int để lưu năm sản xuất của xe (ví dụ 2018)
- make. Một biến string lưu tên hãng sản xuất. (ví dụ Toyota)
- model. Một biến string lưu tên model (mẫu xe ví dụ Camry)
- speed. Tốc độ xe chạy
- color Một biến kiểu string lưu màu sơn xe (ví dụ red)

Ngoài ra, lớp nên có các hàm thành phần sau đây.

- Hàm tạo nhận 4 tham số và gán các giá trị này cho 4 biến thành viên year, make, model và color. Các hàm tạo khởi tạo biến thành viên speed là 0.
- Các hàm truy nhập get và set cho các dữ liệu thành phần.
- Hàm accel() để tăng tốc độ xe chạy. Hàm có tác dụng cộng thêm 5 vào biến thành phần speed mỗi lần nó được gọi.
- Hàm brake() để phanh và giảm tốc: Hàm có tác dụng trừ 5 từ biến thành phần speed mỗi lần nó được gọi và đảm bảo speed không < 0

Thể hiện hoạt động của lớp Car trong chương trình bằng cách tạo 2 đối tượng car1 và car2 và sau đó gọi hàm tăng tốc accel() 5 lần trên car1, 6 lần trên car2. Sau mỗi lần tăng tốc, hãy lấy tốc độ hiện tại của các xe và in ra màn hình. Sau đó, gọi hàm phanh brake() 6 lần trên car1 và 5 lần trên car2. Sau mỗi lần phanh, lấy tốc độ hiện tại của các xe và hiển thị ra màn hình.

II) Phân tích xác định tên lớp và các thành phần, mối quan hệ

Bài 6

Sinh viên đăng ký môn học như sau. Trước khi bắt đầu học kỳ, phòng đào tạo đưa ra một danh sách các lớp tín chỉ sẽ mở (mỗi lớp tín chỉ gồm có mã lớp, tên lớp, tên nhóm và mã môn học, thời gian học) mà sinh viên có thể đăng ký vào học. Mỗi môn học gồm tên môn học, mã môn học, số tín chỉ). Mỗi sinh viên không được đăng ký dưới 15 và quá 30 tín chỉ trong 1 học kỳ. Sau khi chọn lớp đăng ký, mỗi sinh viên có một bản đăng ký gồm có mã sinh viên, học kỳ, năm học và có một danh sách các lớp tín chỉ mà sinh viên đã đăng ký thành công.

- 1) Hãy tìm ra các thực thể chứa dữ liệu trong bài toán trên, sau đó
 - Xác định các tên lớp
 - Với mỗi lớp, xác định các thuộc tính bao gồm tên thuộc tính và kiểu dữ liệu của thuộc tính
 - Với mỗi lớp, xác định các phương thức của lớp
- 2) Xác định mối quan hệ có thể có giữa các lớp theo các mối quan hệ như liên kết truy nhập (use a) , hợp thành (has a) và thừa kế (is a)
- 3) Giả sử rằng nếu cần biết tổng số tín chỉ mà 1 sinh viên đã đăng ký, ta sẽ nhập mã sinh viên và chương trình sẽ in ra tổng số tín chỉ. Vậy sẽ cần đến dữ liệu thành phần của những lớp nào để có thể tính được tổng số tín chỉ mà sinh viên đã đăng ký.

III) Lập trình giải quyết vấn đề theo kiểu lập trình hướng đối tượng

Bài 7

Lập chương trình quản lý danh bạ thông tin đối tác của 1 công ty với mỗi mục bao gồm: tên đối tác, cơ quan công tác, chức vụ, năm sinh, ngày sinh, số điện thoại di động, số điện thoại cơ quan, địa chỉ nhà riêng, địa chỉ cơ quan, địa chỉ email, ghi chú...

- a) Nhập dữ liệu cho danh bạ (trong danh bạ lưu được tối đa 500 đối tác)
- b) In danh bạ ra màn hình ở dạng bảng với các cột là các trường thông tin như STT, tên, cơ quan, chức vụ, số di động, email
- c) Tìm kiếm đối tác trong danh bạ theo tên và cơ quan. Nếu tìm thấy thì in ra màn hình các trường thông tin như trong câu b)

d) Sắp xếp danh bạ theo tên với thứ tự ABC và in ra màn hình với các trường thông tin như trong câu b)

Bài 8.

Thông tin về kết quả học tập môn Ngôn ngữ lập trình C++ của sinh viên bao gồm: điểm chuyên cần (10%) với 10 buổi điểm danh, điểm bài tập (10%) , điểm thực hành (10%), điểm bài kiểm tra giữa kỳ (30%), điểm bài thi cuối kỳ (40%). Mỗi điểm tính riêng theo thang điểm 10.

Lập trình tạo các kiểu dữ liệu struct có thể lưu kết quả học tập của sinh viên trong một lớp với số tối đa là 60

- Nhập tên lớp và số lớp
- Nhập thông tin sinh viên gồm: mã số sv, họ tên cho từng sinh viên trong lớp học
- Với từng sinh viên đã có, nhập số buổi điểm danh và các điểm thành phần của từng sinh viên (Khi nhập thì in ra mã sinh viên và họ tên)
- Với từng sinh viên, tính điểm quá trình biết điểm quá trình là tích lũy của điểm chuyên cần và điểm bài tập và điểm thực hành (chiếm 30% điểm học phần)
- Với từng sinh viên, tính điểm kiểm tra biết điểm kiểm tra là tích lũy của điểm bài kiểm tra giữa kỳ và bài thi cuối kỳ (chiếm 70% điểm học phần)
- Sắp xếp danh sách sinh viên theo điểm học phần giảm dần
- In danh sách kết quả học tập của sinh viên ra màn hình với các cột là các trường dữ liệu: STT, họ tên, mã số sinh viên, điểm quá trình, điểm kiểm tra, điểm học phần (như bảng bên dưới)
- Lọc và chỉ in ra danh sách các sinh viên có điểm học phần ≥ 5.50 theo định dạng của câu g)

Lop: 61PM										
Si so: 2										
STT		Ho ten		MSSV		DQT		DKT		DHP

1		Tran Van Nam		12361		7.5		8.8		8.4
2		Nguyen Van Tuan		45661		6.5		7.0		6.9