



## BÀI THỰC HÀNH SỐ 05

### 1. MỤC TIÊU

Hiểu và cài đặt được chương trình có sử dụng kiểu dữ liệu cấu trúc (struct).

### 2. BÀI TẬP DO SINH VIÊN TỰ THỰC HIỆN

#### Cấu trúc lồng nhau

**Bài 1.** Viết chương trình về hỗn số dưới dạng các hàm:

- Dựa vào cấu trúc phân số đã có ở các bài tập trước, khai báo kiểu dữ liệu để biểu diễn một hỗn số (HONSO), với hỗn số gồm 2 trường (field) là phần nguyên và phần phân số.
- Viết hàm rút gọn 1 hỗn số. void RutGonPS(HONSO &hs)

Trong đó cần thực hiện các xử lý sau:

- Khi tử số lớn hơn mẫu số, cần chuyển phần thương của phép chia giữa tử và mẫu sang cho phần nguyên.
- Khi tử số = 0 thì mẫu số sẽ được gán = 1.
- Khử dấu âm khi tử và mẫu số cùng mang dấu âm.
- Khi tử số là số dương và mẫu số là số âm thì chuyển dấu âm cho tử số (mẫu số luôn mang dấu dương).
- Tối giản hỗn số. Ví dụ:  $1(-9/2)$  sẽ tối giản thành  $-5(1/2)$ .

- Nhập 1 hỗn số. Chương trình sẽ báo lỗi và yêu cầu người dùng nhập lại cho đến khi mẫu số khác 0 là hoàn tất: void Nhap1PS(HONSO &hs)
- Tạo một mảng các hỗn số: void NhapMangPS(HONSO A[], int n). Hàm này sẽ gọi hàm nhập 1 hỗn số ở câu c.
- Xuất 1 hỗn số dưới dạng phần nguyên(tử số/mẫu số), ví dụ  $-3(1/2)$ :

void Xuat1PS(HONSO hs)

- Xuất mảng hỗn số: void XuatMangPS(HONSO A[], int n). Hàm này sẽ gọi hàm xuất 1 hỗn số ở câu e.
- Tính tổng, hiệu, tích, thương hai hỗn số, kết quả trả về 1 hỗn số. Ví dụ:

HONSO Tong2PS(HONSO ps1, HONSO ps2)

- So sánh hai hỗn số: HONSO Tong2PS(HONSO ps1, HONSO ps2)
- Tìm hỗn số lớn nhất, hỗn số nhỏ nhất. Ví dụ:

HONSO TimHonSoMax(HONSO A[], int n)

- Sắp xếp mảng hỗn số tăng dần: void SortMangHS(HONSO A[], int n)
- Cho biết tổng, hiệu, tích, thương của 2 hỗn số lớn nhất và hỗn số nhỏ nhất đang có trong mảng.

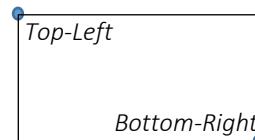


**Bài 2.** Viết chương trình dưới dạng các hàm để quản lý mảng thông tin về sinh viên. Trong đó ngoài những thành phần có kiểu dữ liệu chuẩn của C/C++, cấu trúc sinh viên còn có 2 thành phần có kiểu struct khác là ngày sinh và điểm số.

- a.- Khai báo cấu trúc DATE gồm 3 số nguyên đại diện cho ngày, tháng và năm.
- b.- Khai báo cấu trúc DIEM gồm 3 số thực là điểm toán, điểm lý, điểm tin.
- c.- Khai báo cấu trúc sinh viên, biết thông tin về một sinh viên gồm có:
  - Mã số sinh viên (chuỗi 10 ký tự).
  - Họ Tên (là chuỗi tối đa 30 ký tự);
  - Ngày sinh theo kiểu struct DATE (gồm 3 thành phần kiểu số nguyên là ngày tháng năm).
  - Giới tính (Nam hoặc Nữ).
  - Điểm số theo kiểu struct DIEM.
- d.- Viết hàm nhập một ngày (void Nhap1Ngay(DATE &d)).
- e.- Viết hàm xuất một ngày dưới dạng day/month/year (void Xuat1Ngay(DATE d)). Viết hàm nhập dữ liệu cho một sinh viên (void Nhap1SV(SINHVIEN &sv)).
- f.- Viết hàm nhập danh sách sinh viên, lưu trên mảng một chiều. Hàm này sẽ gọi hàm nhập 1 sinh viên. (NhapMangSV(SINHVIEN A[], int n)).
- g.- Viết hàm xuất dữ liệu một sinh viên với thông tin vừa nhập ở trên (void Xuat1SV(SINHVIEN &sv)).
- h.- Viết hàm xuất danh sách sinh viên. (XuatMangSV(SINHVIEN A[], int n)).
- i.- Xuất thông tin của sinh viên có mã sinh viên là “X”.
- j.- Đếm số lượng sinh viên có năm sinh là 2000.
- k.- Sắp xếp danh sách sinh viên theo MSSV
- l.- Xóa sinh viên có mã số “SV123”.

**Bài 3.** Có 5 đối tượng trong mặt phẳng sẽ có trong cùng một chương trình cần viết như sau:

- Điểm (pixel): bao gồm tọa độ x, tọa độ y.
- Đoạn thẳng (line segment): được xác định bởi 2 điểm (x1, y1) là điểm bắt đầu – (x2, y2) là điểm kết thúc.
- Đường thẳng (line): là đoạn thẳng nhưng được mở rộng theo cả 2 hướng, tức là không còn điểm đầu và điểm cuối. Tất cả các điểm trên cùng 1 đường thẳng đều thỏa chung cùng 1 phương trình đường thẳng là  $y=ax+b$ .
- Hình chữ nhật (rectangle): được xác định bởi 2 điểm Top-Left(x1, y1) và Bottom-Right(x2, y2).
- Hình tròn (circle): được xác định bởi một điểm O(x, y) là tâm; và R là bán kính. Phương trình tổng quát của đường tròn (Hệ tọa độ Descartes) là:  $(x - a)^2 + (y - b)^2 = R^2$ .





**Yêu cầu thực hiện:**

- Hãy định nghĩa cấu trúc của các kiểu dữ liệu cho 4 đối tượng trên mà sinh viên cho là hợp lý nhất.
- Cho nhập 2 phương trình đường thẳng. Tìm giao điểm của 2 đường thẳng.

**Nhắc lại:**

- *Khi hai đường thẳng giao nhau:* chúng chỉ gặp nhau tại một điểm được thể hiện bằng cặp tọa độ  $x$  và  $y$ . Vì cả hai đường thẳng đều đi qua điểm đó nên cặp tọa độ  $x, y$  phải thỏa mãn cả hai phương trình. Để tìm giao điểm này, ta cho vế phải của 2 phương trình bằng nhau để tìm  $x$ . Từ  $x$  có được, thay vào phương trình để tìm  $y$ . Ví dụ:

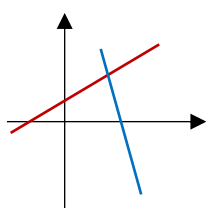
Phương trình đường thẳng 1:  $y = x + 3$

Phương trình đường thẳng 2:  $y = -2x + 15$

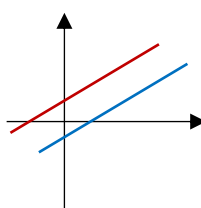
Để tìm  $x$ : cho 2 vế phải bằng nhau:  $x + 3 = -2x + 15$

Giải phương trình bậc 1 này để tìm  $x$ .

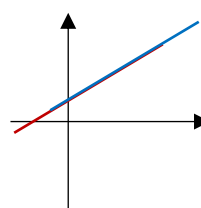
- *Khi hai đường thẳng không giao nhau:*
  - Trường hợp hai đường thẳng song song: khi đó, các số hạng  $x$  sẽ bị triệt tiêu và phương trình được đơn giản thành một mệnh đề sai (chẳng hạn  $0=1$ ). Đáp án lúc này là "*hai đường thẳng không giao nhau*" hoặc "*không có nghiệm thực*".
  - Trường hợp hai phương trình biểu diễn cùng một đường thẳng: khi đó chúng "giao nhau" ở mọi điểm. Các số hạng  $x$  sẽ bị triệt tiêu và phương trình được đơn giản thành một mệnh đề đúng (chẳng hạn  $2=2$ ). Đáp án của trường hợp này là "*hai đường thẳng trùng nhau*".



2 đường thẳng giao nhau

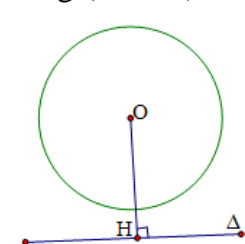


2 đường thẳng song song

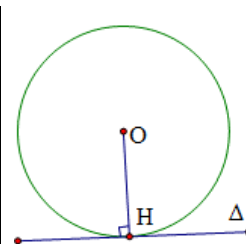


2 đường thẳng trùng nhau

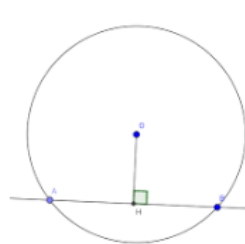
- Cho nhập 1 phương trình đường thẳng, 1 phương trình đường tròn. Tìm giao điểm của chúng (nếu có).



SL giao điểm = 0



SL giao điểm = 1



SL giao điểm = 2

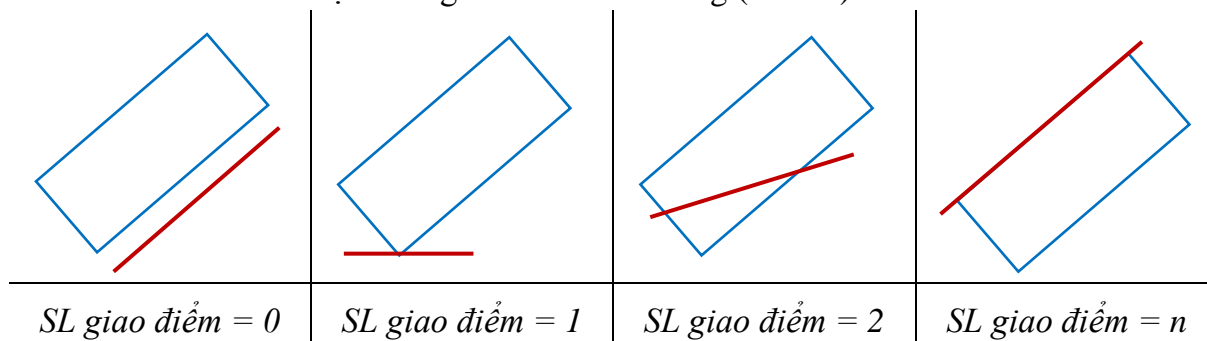


➤ Nhắc lại: Giả sử cho phương trình đường thẳng:  $y = x + 7$   
Và phương trình đường tròn:  $y = x^2 + 2x + 1$

Thực hiện qua các bước

- Cho vế phải của 2 phương trình bằng nhau ( $x + 7 = x^2 + 2x + 1$ ).
- Dựa kết quả vừa có ( $x^2 + x - 6$ ), giải phương trình bậc 2 để tìm nghiệm. Nghiệm có được chính là giao điểm cần tìm.

d. Cho nhập 1 phương trình đường thẳng và tọa độ 2 điểm Top-Left và Bottom-Right của 1 hình chữ nhật. Tìm giao điểm của chúng (nếu có).



### Con trỏ và kiểu dữ liệu có cấu trúc

**Bài 4.** Tổ chức dữ liệu quản lý danh mục các xe máy của một cửa hàng. Các thông tin liên quan đến mỗi xe gồm:

- Mã model (kiểu số nguyên).
- Model (kiểu chuỗi, tối đa 10 ký tự).
- Hãng sản xuất (kiểu chuỗi, tối đa 30 ký tự).
- Lắp ráp tại Việt Nam (kiểu bool). Quy ước: true-Việt Nam; false-nước ngoài.
- Giá bán (kiểu float). Quy ước giá bán  $>0$
- Thời gian bảo hành (kiểu số nguyên). Quy ước: **1**: 1 năm; **3**: 3 năm; **5**: 5 năm.
- Số máy (kiểu chuỗi, tối đa 8 ký tự).
- Số khung (kiểu chuỗi, tối đa 12 ký tự).
- Màu sơn (kiểu chuỗi, tối đa 20 ký tự).

Viết chương trình sử dụng con trỏ để viết các hàm chức năng thực hiện những công việc sau:

- Khai báo cấu trúc Model gồm mã model, tên model, tên hãng sản xuất, lắp ráp tại Việt Nam, giá bán, thời gian bảo hành. Cần kiểm tra giá trị nhập cho các trường (field, thuộc tính), nếu nhập sai, chương trình phải yêu cầu nhập lại cho đến khi nhập đúng:
  - Lắp ráp tại Việt Nam: cho người dùng nhập Y hoặc y là Lắp ráp tại Việt Nam; nhập N hoặc n là Lắp ráp tại nước ngoài. Các ký tự được nhập phải là 1 trong các ký tự Y, y, N, n.
  - Thời gian bảo hành chỉ nhận 1 trong 3 giá trị 1 hoặc 3 hoặc 5.



- b. Khai báo cấu trúc Xe gồm các thuộc tính còn lại và thuộc tính Mã model.
  - c. Hàm nhập vào 1 Model mới cùng với các thông tin liên quan đến Model này.
  - d. Hàm nhập vào 1 xe mới cùng với các thông tin liên quan đến xe này.
  - e. Hàm nhập danh sách n xe, với n do người dùng nhập trực tiếp trong hàm.
  - f. Hàm xuất thông tin về 1 xe. Yêu cầu thuộc tính Lắp ráp xuất tại Việt Nam: phải ghi rõ “*Lắp ráp tại Việt Nam*” hoặc “*Lắp ráp tại nước ngoài*”.
  - g. Hàm xuất các xe đang có trong danh sách.
  - h. Hàm cho nhập hãng sản xuất: In ra danh sách các xe do hãng này sản xuất gồm tên model, số máy, số khung, màu sơn.
  - i. Hàm nhận 1 tham số là số khung, cho biết tên hãng sản xuất và tên model của xe.
  - j. Hàm nhận 2 tham số nguyên là giá X và tên hãng Y. Hàm thực hiện in ra danh sách các xe lắp ráp tại Việt Nam sản xuất, của hãng X và có giá thấp hơn Y.
-