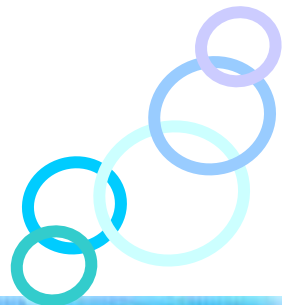


Phương pháp Lập trình hướng đối tượng

Nguyễn Khắc Huy

BMCNPM – ĐHKHTN TPHCM
09/2014



Giới thiệu môn học

- Lập trình hướng đối tượng
 - Tên tiếng Anh: Object Oriented Programming
 - Số tín chỉ: 4
- Trao đổi thảo luận
 - Diễn đàn
 - ✓ PP Lập trình hướng đối tượng - 13CTT11
 - ✓ <http://courses.fit.hcmus.edu.vn/moodlenew>

Giới thiệu môn học

□ GV Lý thuyết

- Nguyễn Khắc Huy

- ✓ nkhuy@fit.hcmus.edu.vn

- ✓ Phòng làm việc: I82, NVC

□ GV Thực hành & Trợ giảng

- Ngô Chánh Đức

- ✓ ncduc@fit.hcmus.edu.vn

- Nguyễn Lê Hoàng Dũng

- ✓ nlhdung@fit.hcmus.edu.vn

Qui định môn học

- Chăm chỉ làm đầy đủ các bài tập.
- Đọc thêm các tài liệu tham khảo bằng tiếng Anh.
- Học tập chủ động và có tinh thần tự giác.

Hệ thống điểm

□ Lý thuyết: 70%

○ Bài tập hàng tuần 10%

○ Kiểm tra giữa kỳ 20%

○ Thi viết cuối khóa 40%

○ Năng nổ tham gia hoạt động lớp 5%

NEW

□ Thực hành: 30%

○ Thông báo sau

□ Tổng điểm (LT + TH) $\geq 5 \rightarrow$ **đậu**

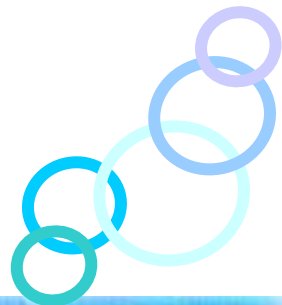
□ Dưới 5 điểm hoặc điểm TH=0 \rightarrow **rớt**

Mục tiêu và nội dung

- Giới thiệu sinh viên phương pháp lập trình mới: PPLTHĐT
- Thay đổi tư duy lập trình từ lập trình hàm-thủ tục sang lập trình hướng đối tượng
- Sử dụng ngôn ngữ C++ để thể hiện
- Sau khi học xong, sinh viên có thể tự thiết kế và cài đặt giải quyết các bài toán bằng phương pháp lập trình hướng đối tượng

Các chủ đề

- Tổng quan và giới thiệu PPLTHĐT
- Đối tượng, lớp và các tính chất
- Hàm tạo và hàm hủy
- Các toán tử
- Kế thừa
- Đa xạ
- Đa kế thừa, lớp trừu tượng
- STL, Templates
- Một số vấn đề nâng cao



Nội dung

- **Chuẩn và quy ước lập trình**
- Ôn tập hàm, cấu trúc, con trỏ.
- Tổng quan về lập trình hướng đối tượng

Chuẩn và quy ước lập trình

□ Vì sao phải có chuẩn và quy ước?

✓ Làm việc một mình:

- Tự làm tự hiểu.
- **Mình luôn hiểu mình?**

✓ Làm việc nhóm:

- Mỗi người một việc.
- Ráp nối công việc.
- **Mọi người luôn hiểu nhau?**

Phối hợp công việc hiệu quả

Áp đặt kỷ luật!!

Chuẩn và quy ước lập trình

- **Không có chuẩn chung toàn thế giới!!**
- Quy ước đặt tên (Naming Convention):
 - ✓ Quy tắc vàng: tên phải thể hiện ý nghĩa.
 - x, y, f, g, ... ?!
 - total, rate, create, run, ... !!
 - ✓ Quy tắc viết tên theo kiểu “lạc đà” (Camel Case)
 - Dùng để viết các từ dính liền nhau.
 - Viết hoa chữ cái đầu mỗi từ.
 - **U**pper**C**amel**C**ase.
 - **l**ower**C**amel**C**ase.

Chuẩn và quy ước lập trình

□ Quy ước viết câu lệnh:

- ✓ Quy tắc vàng: viết câu lệnh rộng rãi, rõ ràng.

- `x=a+b-c*d; for(int i=0;i<n;i++);`

- `x = a + b - c * d;`

- `for (int i = 0; i < n; i++);`

- ✓Viết mỗi câu lệnh một dòng.

- ✓Viết cách khoảng giữa hai đoạn lệnh.

□ Quy ước viết chú thích:

- ✓ Quy tắc vàng: viết chú thích đầy đủ, dễ hiểu.

- ✓Viết chú thích cho từng hàm.

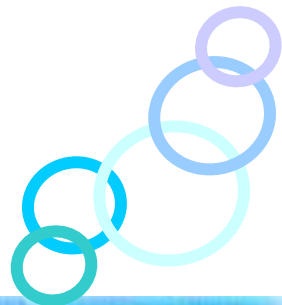
- ✓Dùng dấu `//` thay cho `/* */`.

Nội dung

- Chuẩn và quy ước lập trình
- **Ôn tập hàm, cấu trúc, con trỏ.**
- Tổng quan về lập trình hướng đối tượng

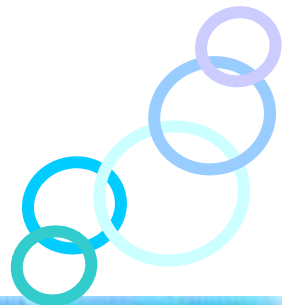
Ôn tập hàm, cấu trúc, con trỏ

- ❑ Chương trình là gì?
- ❑ Hàm là gì? Tham số? Có mấy cách truyền?
- ❑ Biến là gì? Hằng là gì?
- ❑ Cấu trúc là gì? Cách sử dụng? Ý nghĩa?
- ❑ Con trỏ là gì? Cách sử dụng?
- ❑ Nhập xuất trong C++?
- ❑ Header File để làm gì?



Ôn tập hàm, cấu trúc, con trỏ

- Hàm, cấu trúc, con trỏ:
 - Hàm chứa xử lý của chương trình.
 - Cấu trúc chứa dữ liệu của chương trình.
 - Cấu trúc dùng để đóng gói dữ liệu.
 - Con trỏ chứa địa chỉ của biến hoặc vùng nhớ.
 - Dùng con trỏ để thao tác linh động.



Bài tập hàm & thủ tục

□ Bài tập 1.1:

Viết chương trình cho phép thực hiện các thao tác trên kiểu **phân số**:

- ✓ Nhập, xuất phân số.
- ✓ Nghịch đảo, rút gọn phân số.
- ✓ Cộng, trừ, nhân, chia hai phân số.

Bài tập hàm & thủ tục

□ Bài tập 1.2:

Viết chương trình cho phép thực hiện các thao tác trên kiểu **số phức**:

- ✓ Nhập, xuất số phức.
- ✓ Tính module số phức.
- ✓ Cộng, trừ, nhân, chia hai số phức.

Bài tập hàm & thủ tục

□ Bài tập 1.3:

Viết chương trình cho phép thực hiện các thao tác trên kiểu **đơn thức**:

- ✓ Nhập, xuất đơn thức.
- ✓ Tính giá trị, đạo hàm, nguyên hàm đơn thức.
- ✓ Cộng, trừ, nhân chia hai đơn thức cùng bậc.

Bài tập hàm & thủ tục

□ Bài tập 1.4:

Thông tin một học sinh bao gồm:

- ✓ Họ tên.
- ✓ Điểm văn, toán.

Viết chương trình cho phép thực hiện các thao tác trên kiểu **học sinh**:

- ✓ Nhập, xuất thông tin học sinh.
- ✓ Tính điểm trung bình.
- ✓ Xếp loại theo tiêu chí
 - Giỏi (≥ 8.0), Khá (≥ 7.0).
 - Trung bình (≥ 5.0), Yếu (< 5).

Bài tập hàm & thủ tục

□ Bài tập 1.5:

Viết chương trình cho phép thực hiện các thao tác trên kiểu **mảng**:

- ✓ Nhập, xuất mảng.
- ✓ Lấy kích thước mảng.
- ✓ Lấy phần tử tại vị trí nào đó.
- ✓ Tìm phần tử nào đó trong mảng.
- ✓ Sắp xếp tăng, giảm.

Nội dung

- Chuẩn và quy ước lập trình
- Ôn tập hàm, cấu trúc, con trỏ.
- **Tổng quan về lập trình hướng đối tượng**

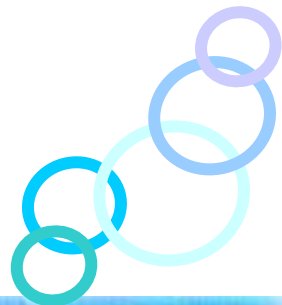


Phần mềm

- Viết 1 phần mềm bao gồm rất nhiều công đoạn: phân tích & thiết kế, cài đặt, kiểm tra/thử nghiệm và bảo trì.
- Cài đặt (programming/coding) chỉ là 1 phần trong quá trình trên.
- Lập trình hướng đối tượng là một cách tiếp cận (hay phương pháp/kiểu) lập trình.
- C++/C#/Java v.v... là ngôn ngữ lập trình

Sự tiến hóa của ngôn ngữ lập trình

- Ngôn ngữ máy (bit, byte v.v...)
- Hợp ngữ
- Ngôn ngữ cấp cao: Pascal, C/C++, C#, Java, Visual C++, Visual Basic, Visual DotNet v.v...
 - Lập trình tuyến tính
 - Lập trình có cấu trúc (thủ tục/hàm...)
 - Lập trình hướng đối tượng
- ...



Bài tập

- Dùng cấu trúc dữ liệu kiểu struct, viết chương trình:
 - struct Diem: mô tả các tọa độ của 1 điểm trong mặt phẳng.
 - struct TamGiac: mô tả 3 đỉnh của tam giác trong mặt phẳng
 - Hàm tính khoảng cách giữa 2 điểm: `khoangCach`
 - Hàm tính chu vi, diện tích của tam giác: `chuVi`, `dienTich`

Lập trình hướng thủ tục

```
struct Diem  
{  
    int x, y;  
};
```

```
float KhoangCach(Diem m, Diem n);
```

```
struct TamGiac  
{  
    Diem x, y, z;  
};
```

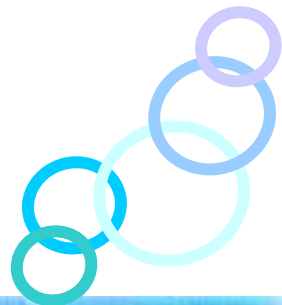
```
float DienTich(TamGiac a);  
float ChuVi(TamGiac a);
```

- Chương trình thường bị trộn lẫn, không rõ ràng, rạch ròi giữa các cấu trúc dữ liệu khác nhau.

→ Khó theo dõi, sửa chữa hay thay đổi khi có vấn đề phát sinh

Hướng đối tượng

- Hiểu được “hướng đối tượng” không phải thông qua cách học 1 ngôn ngữ hay một bộ công cụ nào đó
- ➔ Mà đó là 1 cách tiếp cận, 1 cách suy nghĩ khác

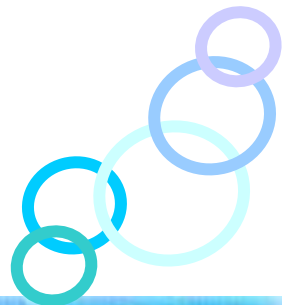


Vấn đề!

- Học một ngôn ngữ lập trình mới là rất quan trọng.
Tuy nhiên việc hiểu ý nghĩa của 1 khái niệm, 1 cách suy nghĩ mới còn quan trọng hơn gấp nhiều lần
 - Cụ thể, một số người lập trình tự nhận mình là người lập trình C++ nhưng thật ra họ chỉ là người lập trình C nhưng dùng C++ compiler
 - Học UML trước khi học khái niệm hướng đối tượng cũng tương tự như chúng ta đọc 1 bản mạch điện mà không có kiến thức về điện

Ng nghiêm trọng hơn!!!

- Vấn đề còn trở nên nghiêm trọng hơn nữa khi một người lập trình sử dụng một số tính năng của lập trình hướng đối tượng để giải quyết vấn đề mà người lập trình hướng đối tượng không thể hiểu nổi

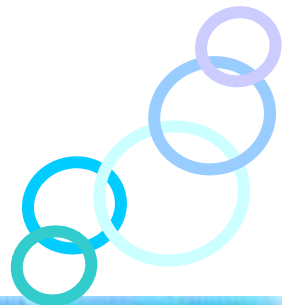


Lập trình hướng đối tượng

Lập trình hướng đối tượng là phương pháp lập trình dựa trên kiến trúc **lớp** (class) và **đối tượng** (object)

Đối tượng (object) là gì?

- Đối tượng trong thế giới thực: là một thực thể cụ thể mà ta có thể sờ, nhìn thấy hay cảm nhận được.
- Đối tượng phần mềm: dùng để biểu diễn các đối tượng trong thế giới thực.
- Mỗi đối tượng bao gồm 2 thành phần: thuộc tính và hành động.



Ví dụ: một người

- Một người có các thuộc tính: tên, tuổi, địa chỉ, màu mắt...
- Các hành động: đi, nói, thở...

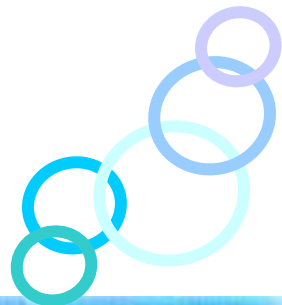
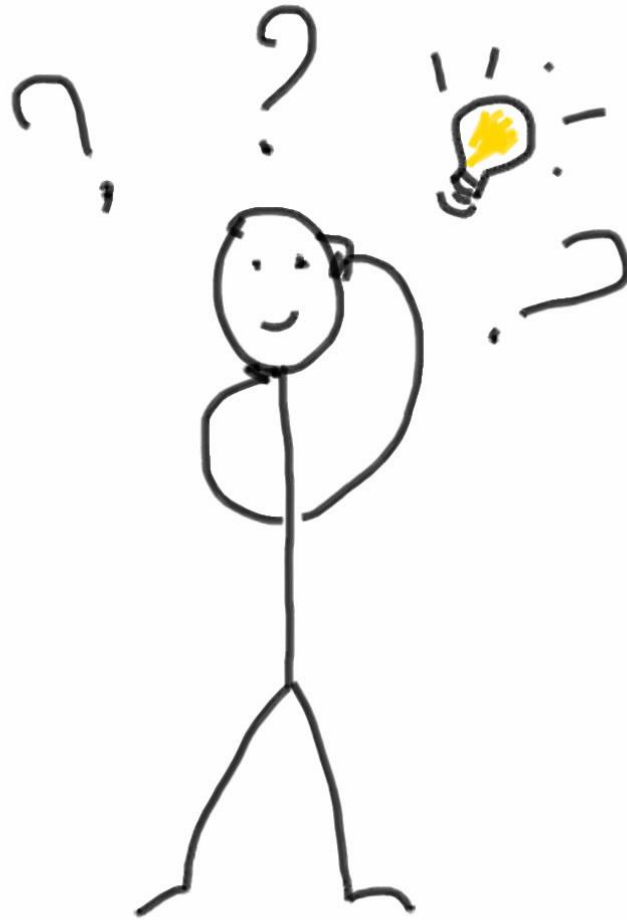
**Một đối tượng là 1 thực thể bao gồm
thuộc tính và hành động**

Xác định đối tượng

□ Ví dụ:

- Con mèo của cô hàng xóm có bộ lông màu vàng.
- Cây cổ thụ giữa sân trường em đã nhiều tuổi lắm rồi.
- Mèo là động vật có bốn chân.
- Quán cơm bà Bảy hôm nay không có một người khách.
- Sinh viên trường ĐH KHTN rất năng động.

Đối tượng và lớp



Tài liệu tham khảo

- Stephen Prata, **C++ Primer Plus**, 4th Edition, *Sams*, 2004.
- Bjarne Stroustrup, **The C++ Programming Language**, 4th Edition, Addison-Wesley, 2013.
- Marshall Cline, **C++ FAQ Lite**, 2nd Edition, Addison-Wesley, 2000.
- Robert L. Kruse, Alexander J. RybaData, **Data Structures and Program Design in C++**, *Prentice-Hall, Inc.*, 2000
- Bruce Eckel, **Thinking in C++**, *Prentice Hall*, 1998.

Tài liệu tham khảo

- Trần Đan Thư, Giáo trình hướng đối tượng Khoa CNTT, trường ĐHKHTN.
- Phạm Văn Ất, **C++ và Lập trình hướng đối tượng**, NXB Khoa Học Kỹ Thuật, 1999.
- Trần Văn Lăng, **Lập trình hướng đối tượng C++**, NXB Thống Kê, 1997.

Lời cảm ơn

- Nội dung được xây dựng dựa trên slide trình bày của Thầy Đinh Bá Tiến