



# GIỚI THIỆU

LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

GVHD: Trương Toàn Thịnh

{ [ttthinh@fit.hcmus.edu.vn](mailto:ttthinh@fit.hcmus.edu.vn) }

# NỘI DUNG

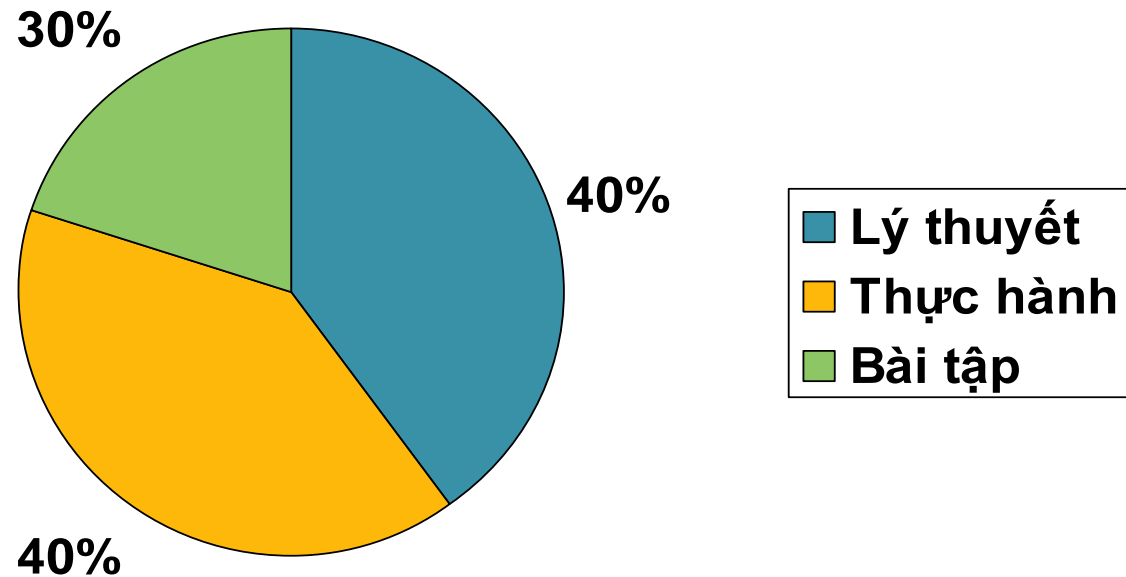
- Mục tiêu & nội dung môn học
- Các chuẩn và qui ước lập trình
- Ôn tập
  - Hàm
  - Con trỏ
  - Cấu trúc
- Bài tập

# MỤC TIÊU & NỘI DUNG

- **Định nghĩa được** các khái niệm cơ bản của Object Oriented Programming.
- **Áp dụng được** các tính chất của OOP để giải quyết vấn đề **bằng C++**.
- Môi trường thực hành:
  - Visual C++.NET 2013 → 2017.

# MỤC TIÊU & NỘI DUNG

- Thang điểm<sup>2</sup>



# MỤC TIÊU & NỘI DUNG

- Tổ chức lớp học
  - Sinh viên tổ chức thành nhóm 4 người:
    - Thảo luận.
    - Làm bài tập.
  - Mỗi sinh viên: đọc thêm các tài liệu tham khảo.
- Trao đổi thảo luận:
  - Diễn đàn: <http://courses.fit.hcmus.edu.vn>
  - GVLT: Trương Toàn Thịnh
    - Email: [ttthinh@fit.hcmus.edu.vn](mailto:ttthinh@fit.hcmus.edu.vn).
    - Phòng làm việc: Phòng I82, BM CNPM, Khoa CNTT, Trường ĐH KHTN TP.HCM (hẹn trước).

# MỤC TIÊU & NỘI DUNG

- Tài liệu tham khảo:
  - **C++ và lập trình hướng đối tượng** – Thầy Phạm Văn Ất
  - **Lập trình hướng đối tượng** – Thầy Trần Đan Thư, Thầy Đinh Bá Tiến và Thầy Nguyễn Tấn Trần Minh Khang
  - **The C++ Programming Language** – Bjarne Stroustrup
  - **C++ Primer Plus** – Stephen Prata

# MỤC TIÊU & NỘI DUNG

- Nội dung chính môn học
  - Lớp & đối tượng
  - Các lớp dựng sẵn (built-in classes)
  - Thuộc tính & phương thức
  - Tính chất kế thừa
  - Tính chất đa hình
  - Môi liên hệ
  - Lập trình tập tin
  - Tham số hóa
  - Các thư viện thông dụng
  - Xử lý lỗi & ngoại lệ

# NỘI DUNG

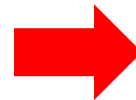
- Mục tiêu & nội dung môn học
- Các chuẩn và qui ước lập trình
- Ôn tập
  - Hàm
  - Con trỏ
  - Cấu trúc
- Bài tập



# CÁC CHUẨN & QUY ƯỚC

- 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!!**

# CÁC CHUẨN & QUY ƯỚC

- 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, ... !!
  - Đặt tên theo kiểu “lạc đà” (Camel Case)
    - Dùng để viết các từ dính liền nhau.
    - Viết hoa chữ cái đầu mỗi từ.
    - UpperCamelCase.
    - lowerCamelCase.



# CÁC CHUẨN & QUY ƯỚC

- 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

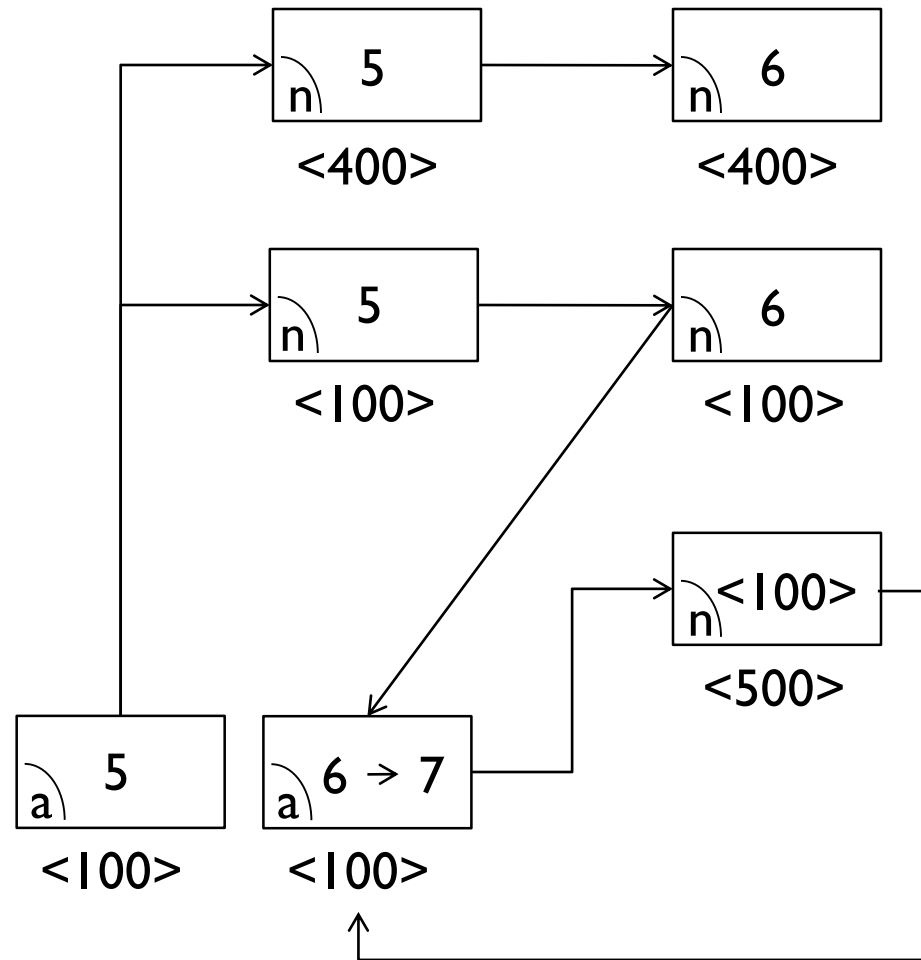
- Mục tiêu & nội dung môn học
- Các chuẩn và qui ước lập trình
- Ôn tập
  - Hàm
  - Con trỏ
  - Cấu trúc
- Bài tập

# ÔN TẬP – HÀM

- Hàm là nơi chứa xử lý của chương trình
- Có 3 cách truyền tham số cho hàm
  - Tham trị
  - Tham chiếu
  - Tham con trỏ
- Con trỏ hàm là biến chứa địa chỉ của hàm
- Hàm trùng tên, hàm có giá trị mặc định (giá trị mặc định **PHẢI** nằm ở cuối danh sách tham số)

# ÔN TẬP – HÀM

```
void increaseValue(int n){
    n++;
}
void increaseRef(int& n){
    n++;
}
void increasePnt(int* n){
    (*n)++;
}
void main(){
    int a = 5;
    increaseValue(a);
    cout << a << endl;
    increaseRef(a);
    cout << a << endl;
    increasePnt(&a);
    cout << a << endl;
}
```



# ÔN TẬP – CON TRỎ

- Con trỏ là nơi chứa địa chỉ ô nhớ
- Truyền tham số dạng con trỏ dạng
  - Tham chiếu
  - Tham trị
- Con trỏ hàm chứa địa chỉ của hàm: ví dụ câu lệnh khai báo con trỏ hàm tên f trỏ tới các hàm có nguyên mẫu hàm với đầu vào là float và đầu ra là float:

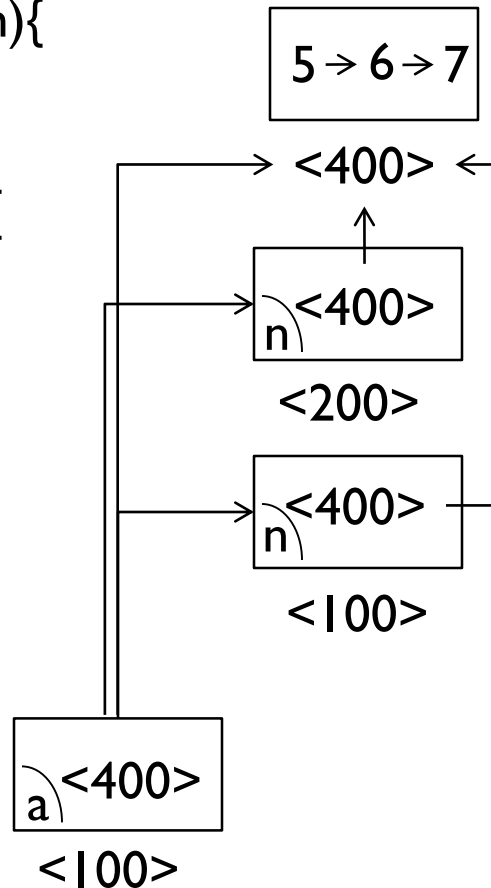
“float (\*f)(float);”

# ÔN TẬP – CON TRỎ

```
void increasePntRef(int* &n){
    (*n)++;
}

void increasePntVal(int* n){
    (*n)++;
}

void main(){
    int *a = new int;
    *a = 5;
    increasePntVal (a);
    cout << *a << endl;
    increasePntRef(a);
    cout << *a << endl;
}
```





# ÔN TẬP – CON TRỎ

```
int _min(int a, int b) {  
    if(a < b) return a;  
    return b;  
}  
  
int _max(int a, int b) {  
    if(a > b) return a;  
    return b;  
}  
  
int _get(int a, int b, int (*s)(int, int)) {  
    return s(a, b);  
}  
  
void main() {  
    int a = _get(3, 5, _max);  
    int b = _get(3, 5, _min);  
    cout << a << endl;  
    cout << b << endl;  
}
```

# ÔN TẬP – CẤU TRÚC

- Khai báo kiểu cấu trúc

```
struct phanso{  
    long tu, mau;  
};
```

```
typedef struct phanso PHANSO;
```

- Khai báo và định nghĩa biến kiểu cấu trúc  
PHANSO ps = {2, 3};
- Gán hai biến cấu trúc không có trường con  
trở

```
PHANSO ps1 = {2, 3}, ps2;  
ps2 = ps1;
```

# ÔN TẬP – CẤU TRÚC

- Khai báo kiểu cấu trúc có trường kiểu mảng

```
struct hocsinh{  
    char hoTen[30];  
    float diemTB;  
};
```

```
typedef struct hocsinh HOCSINH;
```

- Gán hai biến kiểu cấu trúc có trường kiểu mảng

```
HOCSINH hs1 = {“le thi luu”, 9.0}, hs2;  
hs2 = hs1;
```

# NỘI DUNG

- Mục tiêu & nội dung môn học
- Các chuẩn và qui ước lập trình
- Ôn tập
  - Hàm
  - Con trỏ
  - Cấu trúc
- Bài tập

# Bài tập

- Bài tập 1.1
  - Viết chương trình thực hiện các thao tác trên kiểu **phân số**:
    - Nhập, xuất phân số.
    - Nghịch đảo, rút gọn phân số.
    - Phép toán:  $+-\times\div$
- Gợi ý:
  - Tìm ước số chung lớn nhất của hai số a & b:  
USCLN(a, b)  
**IF** b == 0 **THEN RETURN** a;  
**RETURN** USCLN(b, a % b);

# Bài tập

- 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.
- Gợi ý:
  - Gồm 2 phần thực & ảo (**float**)
  - Chia 2 số phức:  $\frac{c + di}{a + bi} = \frac{ac + bd}{a^2 + b^2} + \frac{ad - bc}{a^2 + b^2}i$
  - Tính module cho  $a + bi \rightarrow \sqrt{a^2 + b^2}$

# Bài tập

- 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.
- Gợi ý:
  - Ví dụ đơn thức  $2x^2$ , đạo hàm  $(2x^2)' = 4x$ , nguyên hàm  $\int 4x = 2x^2$ .
  - Công thức nguyên hàm  $\int (ax+b)^\alpha dx = \frac{1}{a} \left( \frac{ax+b}{\alpha+1} \right)^{\alpha+1} + c, \alpha \neq -1$

# Bài tập

- 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 ( $\geq 8.0$ ), Khá ( $\geq 7.0$ ).
      - Trung bình ( $\geq 5.0$ ), Yếu ( $< 5$ ).



# Bài tập

- 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.
- Gợi ý:

```
#define MAX 100
struct Array{
    int a[MAX];
    int size;
};
typedef struct Array ARRAY;
```