

# Lab 10

## Khuôn mẫu hàm

### Lập trình hướng đối tượng

<b>Mục tiêu</b>	Làm quen với việc cài đặt khuôn mẫu hàm đơn giản
-----------------	--

# 1 Hướng dẫn khởi đầu

## Mô tả bài tập

Cài đặt hàm tìm số nhỏ nhất giữa hai số thuộc các loại: số nguyên, số thực, Phân số sử dụng function template

## Hướng dẫn cài đặt

### Bước 1: Tạo mới dự án

- Chọn loại dự án là **C++ / Console Application**.
- Đặt tên solution là: **FunctionTemplate**. Đặt tên project là **SimpleTemplate**
- Nếu sử dụng Visual Studio 2017 trở lên cần **vô hiệu hóa Precompiled header** bằng cách nhấn phải vào project chọn Properties. Vào mục **C / C++ > All Options**, tìm tới tùy chọn **Precompiled header** và chọn **Not using precompiled headers**.

### Bước 2: Tạo mới tập tin Compare.cpp chứa khai báo khuôn mẫu như sau

```
template <class T>
T Min(T a, T b) {
    return a < b ? a : b;
}
```

Chú ý nếu đặt tên là min (viết thường) thì sẽ bị trùng với tên của hệ thống

**Bước 3: Cài đặt hàm main để kiểm tra**

```
int main()
{
    cout << Min(3, 4) << endl;
    cout << Min(3.14, 5.76) << endl;
    cout << Min('A', 'C') << endl;
}
```

## 2 Bài tập vận dụng

### Yêu cầu

1. Cài đặt theo function template để có hàm Max hỗ trợ tìm số lớn hơn và kiểm tra việc cài đặt với loại dữ liệu `int`, `float`, `char`, `Fraction` (phân số - Cần cài đặt toán tử `operator<`)
2. Cài đặt lớp mảng động `DynamicArray` theo function template hỗ trợ kiểu dữ liệu lớp bất kì. Các thao tác cần hỗ trợ:

- + `PushBack(T)` : Thêm một phần tử vào mảng
- + `T GetAt (int)` : Lấy một phần tử tại vị trí truyền vào
- + `T MaxElement()`: Trả ra phần tử lớn nhất của mảng
- + `void Sort()`: hàm sắp xếp tăng mảng

Kiểm tra với các lớp sau: `int`, `float`, `Fraction`, `Point` (so sánh dựa vào khoảng cách tới gốc tọa độ 0-0), `Line` (so sánh dựa trên độ dài) `Rectangle` (so sánh dựa vào diện tích), `Circle` (so sánh dựa vào diện tích), `Triangle` (so sánh dựa vào diện tích).

Chú ý cần nạp chồng toán tử `operator<` hoặc `operator>` tùy theo cách bạn chọn để so sánh.

## 3 Hướng dẫn nộp bài

### Trước khi nộp cần chú ý:

- Lấy tập tin exe được biên dịch sẵn trong thư mục Debug, copy nó ra thư mục Release bên ngoài mã nguồn.
- Xóa hết tất cả các tập tin trung gian trong quá trình biên dịch bằng cách chọn **Build > Clean**.
- Chú ý thư mục ẩn **.vs** rất nặng. Cần hiển thị file ẩn mới thấy và xóa nó đi được.

Nếu bạn muốn biết cách làm đúng thì cần tự tìm cách build ở chế độ **Release** và copy file exe kết quả ra bên ngoài để nộp mới đúng. Tuy nhiên nếu chưa hiểu ý nghĩa thì cứ lấy đại file exe có sẵn đi nộp cũng được (hiện tại đang trong thư mục Debug ứng với chế độ biên dịch Debug)

### Tổ chức bài nộp

- + Thư mục **Source**: chứa mã nguồn đã được clean
- + Thư mục **Release**: chứa tập tin thực thi đã được biên dịch từ mã nguồn
- + Tập tin **readme.txt**: chứa thông tin sinh viên, gồm MSSV và họ tên. Ghi chú kèm các thông tin giáo viên cần chú ý khi chấm bài.

Để nộp bài, nén tất cả lại và đặt tên với định dạng **MSSV.zip** hoặc **MSSV.rar** và nộp.

**Nếu làm đúng các bước trên file này sẽ có kích thước < 100 KB!**

(Tuy nhiên cũng đừng quá lo lắng nếu nó khác con số trên, miễn < 12 MB để nộp được trên moodle là okie nhé!)

**Ngoài lề:** Để đảm bảo sau này nhìn vào file nén còn biết ngay nó làm gì, ta nên thêm vào một số thông tin theo sau MSSV. Ví dụ: 0712221-Lab10-FunctionTemplate.zip. Tuy nhiên việc này là KHÔNG bắt buộc nhé.

-- HẾT --