

# Object-oriented programming

## CS10003

Lecturer: Do Nguyen Kha

# Contents

- Goals
- Standards and Coding Conventions
- Review
- Exercise

# Goals

- Understand concepts of OOP
- Apply OOP to solve problems with C++
- Environments:
  - Visual Studio C++
  - GNU C++
  - or any compatible up to C++20

# Grading

- Final test: 40%
- Team Project: 10%
- Team Seminar of Design Pattern: 10%
- In-Class Exercises: 10%
- Lab: 30%

# Discussion and Contacts

- Facebook Group: <https://www.facebook.com/groups/688834652884919/>
- Lecturer: Đỗ Nguyễn Kha ([dnkha@fit.hcmus.edu.vn](mailto:dnkha@fit.hcmus.edu.vn))
- TA: Nguyễn Lê Hoàng Dũng ([nlhdung@fit.hcmus.edu.vn](mailto:nlhdung@fit.hcmus.edu.vn))
- Lab Instructor: Nguyễn Lê Hoàng Dũng & Mai Anh Tuấn ([matuan@fit.hcmus.edu.vn](mailto:matuan@fit.hcmus.edu.vn))
- Email Subject: [CSC10003][22CLC8] xxx

# References

- **C++ và lập trình hướng đối tượng** – Prof Phạm Văn Ất
- **Lập trình hướng đối tượng** – Prof Trần Đan Thư, PhD Đinh Bá Tiến and PhD Nguyễn Tấn Trần Minh Khang
- **The C++ Programming Language** – Bjarne Stroustrup
- **C++ Primer Plus** – Stephen Prata

# Standards and Coding Conventions

- Why to follow standards and conventions?
  - Working alone:
    - Your code & understand?
    - Always understand yourself?
  - Group working
    - Each person a work to do
    - Reassemble the works
    - Everyone understand each other?

# Standards and Coding Conventions

- No standard for all around the world!!
- Naming convention:
  - Golden rule: name must carry a meaning.
    - x, y, f, g, ... ?!
    - total, rate, create, run, ... !!
  - Naming with “**C**amel **C**ase”
    - Used to write non-separate words
    - Capitalize each character of each word
    - **U**pper**C**amel**C**ase
    - **l**ower**C**amel**C**ase



# Standards and Coding Conventions

- Statement conventions:

- Gold rule: write clearly and enough distance.
  - `x=a+b-c*d; for(int i=0;i<n;i++);`
  - `x = a + b - c * d;`
  - `for (int i = 0; i < n; i++);`
- ONE statement at ONE line
- Separate two coding paragraphs

- Comment conventions:

- Golden rule: write enough and easy-to-understand comments
- Write comments for all functions

# Standards and Coding Conventions

- Google: [https://google.github.io/styleguide/cppguide.html#Header\\_Files](https://google.github.io/styleguide/cppguide.html#Header_Files)
- Chromium:  
<https://chromium.googlesource.com/chromium/src/+/main/styleguide/c++/c++.md>
- Unreal:  
<https://docs.unrealengine.com/5.3/en-US/epic-cplusplus-coding-standard-for-unreal-engine/>
- ...

# Review - Function

- A function is a block of code which only runs when it is called
- You can pass data, known as parameters, into a function
- Functions are used to perform certain actions, and they are important for reusing code: Define the code **once**, and use it **many times**

# Review - Pointer

- A pointer is a variable that stores the **memory address** as its value
- A pointer variable points to a data type (like `int` or `string`) of the same type, and is created with the `*` operator. The address of the variable you're working with is assigned to the pointer
- Using `&` operator to get the address of a variable
- Function pointer is a pointer to a function or memory address of that function  
`int (*func)(int, int);`
- Using `typedef` to define function pointer type  
`typedef int (*myfunc)(int, int);`

# Review - Function Parameter

- Pass by value
- Pass by reference
- Pass by pointer

# Review - Structure

- Structures (also called structs) are a way to group several related variables into one place. Each variable in the structure is known as a member of the structure
- Unlike an array, a structure can contain many different data types (int, string, bool, etc.)

# Team Project

- Members: 4
  - No exception, any team does not have 4 members will be combined with others
- Team formation deadline: 30/09/2023
  - Submit team members list to Google Forms **by team leader**
- Final submission deadline: 1 week **PRIOR** to Final test date (TBD)
  - Progressive submissions are on week 5th, 7th, 9th

# Team Seminar

- Same team as project
- Each team will be assigned a design pattern randomly
- Submissions:
  - PDF report
  - PDF presentation slide
  - Code sample (C++, Java, C#... or any OOP support language)
  - Demo video (YouTube unlisted link)
- Some teams will present on class randomly



Q & A