Lecture #00

Course Introduction

SE271 Object-Oriented Programming (2020)

Yeseong Kim

Original slides from Prof. Shin at DGIST

Overview for Today

- Course Introduction
  - Course Information
  - Objectives
  - Schedule
  - Class Policy
- Very Short Introduction to OOP and C++

Course Information

- Instructor: 김예성
  - Email: yeseongkim@dgist.ac.kr
  - Office: E3-613
    - In-person meeting available in very special cases

- LMS
  - Notice
  - Q&A
  - Scores
  - Homework (tentative)

- TA
  - 임준수 (junsu@dgist.ac.kr)

- (Unofficial but very useful) Textbook
  - C++
    - Learn C++
    - Teach yourself C++ 시리즈
  - OOP Design
    - Design pattern
  - Google!

Course Information

- Office Hour
  - Mon. 12:00~13:00, i.e., right after the class
  - (If you need another time, please email me for an appointment)
  - Happen in Zoom
    - I will post the link in LMS
  - Due to the large volume of the class, please utilize Q&A in LMS for short questions!

- TA Office Hour: Wed. 12:00~13:00

- Utilize the tutoring service as well!
  - Will use Piazza – the link will be available in LMS

This Course for

- (Most likely) Sophomore
  - Have programming experience
  - (assumed) know basic programming concepts incl. control flow, function, variables…
  - (Many of you) want to learn advanced programming

  - Be interactive!

Objectives

- What do you expect to learn?

- In this course,
  - Essential grammar of C++
  - Object-Oriented Programming Concepts (Not ***Objected-Oriented***)
  - Design and implement the programming based on OOP

- Will not cover
  - Basic Programming
  - Complete grammar of C++
    - GUI, system program, …
  - How to use IDE tools (incl. debugger)
  - Advanced Design, e.g., UML

Tentative Schedule

Week	Contents	H/W
1	Course Introduction Hello World!!	
	Variable, Array, basic operator, Comment, Type casting, scope	
2	Flow Control	
	Pointer	
3	Reference type	
	Function (return type, parameters(call by xx), inline)	#1
4	Class (constructor, destructor, vs. Struct)	
	Class (protected, private, public, static method)	
5	Polymorphism 1	#2
	추석 휴일	
6	Polymorphism 2	
	Polymorphism 3	
7	Template 1	#3
	Template 2	
8	Midterm Exam	
9	Inheritance 1	
	Inheritance 2	
10	Virtual function – final(c11)	
	Stream	#4
11	STL	
	STL	
12	Exception Handling	
	Object-oriented Design	
13	Object-oriented Design	
	UML	
14	Team project (Presentation 1)	
	Team project (Presentation 2)	
15	Course Review	
16	Final Exam	

* Schedule is likely to be changed

Assessment

- **Exams (50%)**
  - Midterm (25%)
  - Final exam (25%)
- **Assignments (20%)**
  - 4 HWs (5% each)
  - Late submits : 50% penalty per a day
- **Term project(30%)**
  - 3~4 students per a team
- **Absence penalty**
  - $2^{\max(n - 2, 0)} - 1$

Scores	Grade
100 ~ 85	A
84 ~ 70	B
69 ~ 50	C
50 ~ 0	D

- ** 부정행위는 F 처리할 예정임

Homework Information

- Will use an automated grading tool
  - Don’t try to cheat in any ways
  - Spend your time to understand and implement it by yourself

- Rule of Grace Submission
  - Working condition: When your submission fails to be compiled
  - What to do: Submit an edited version with a report what you change
  - # of Grace(s): *Once* over four homework(s)

* Details will be announced later.

Term Project

- Team Project
  - Team project
    - Any DGIST-related topic chosen by each team
    - A team consists of 3~4 students
  - What you will do
    - Proposal (1 A4 page, font size 12)
    - In-class Presentation
      - Requirements, Class design, etc.
    - Implementation

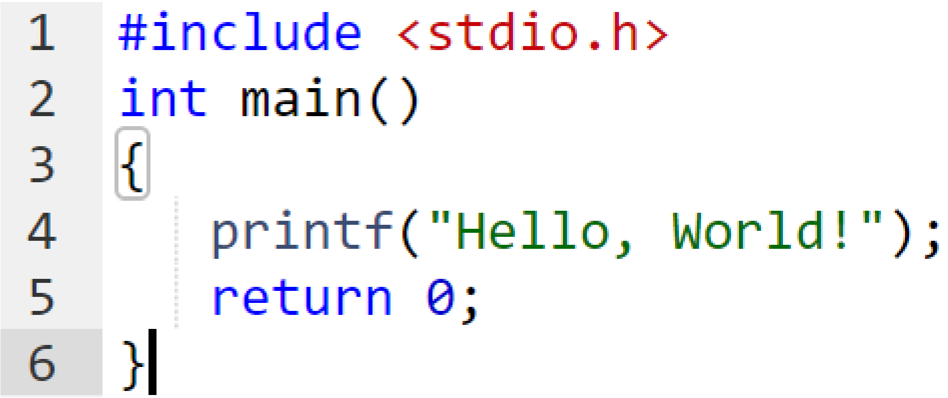
* Details will be announced later.

Very Short Introduction to Object-Oriented Programming

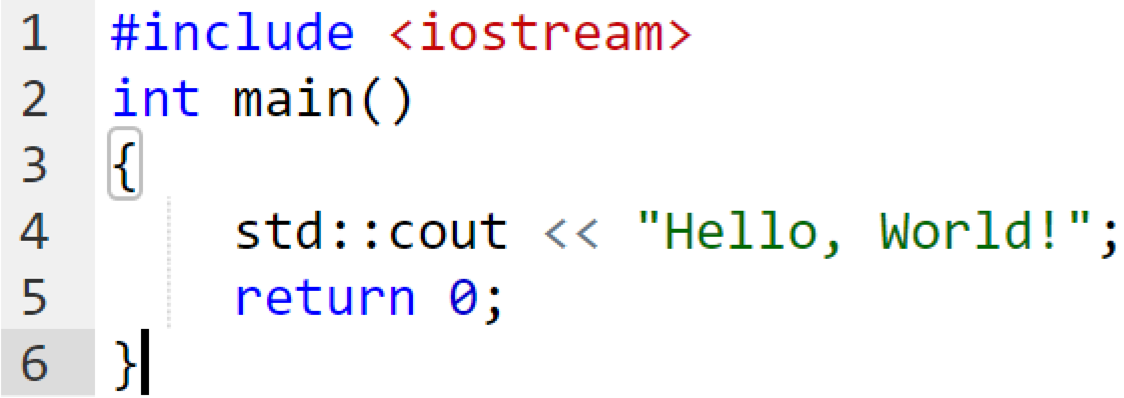
- One of popular programming paradigms
  - Will learn following concepts about **Objects**
    - Abstraction
    - Encapsulation
    - Inheritance
    - Polymorphism
  - cf. Procedural programming (C style)

Short Introduction to C++

- C++ == C ?



```c
#include <stdio.h>
int main()
{
    printf("Hello, World!");
    return 0;
}
```



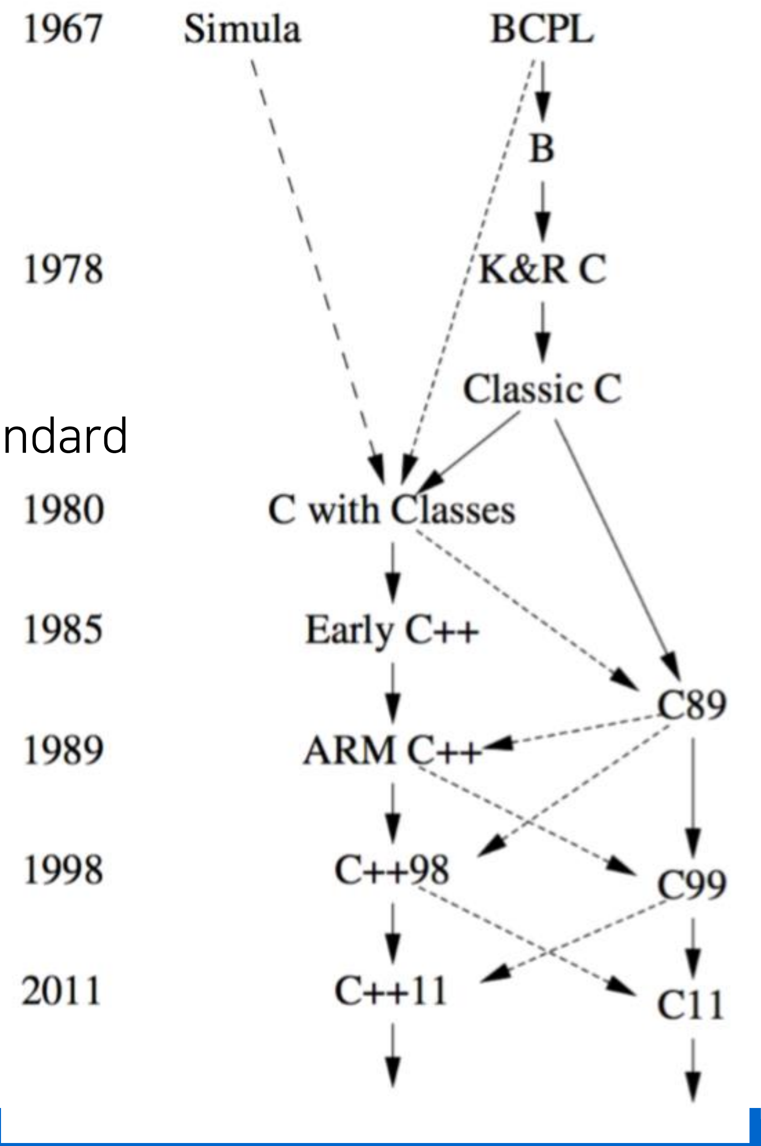
```cpp
#include <iostream>
int main()
{
    std::cout << "Hello, World!";
    return 0;
}
```

- C++ is a superset of C?

History of C/C++

- 1970 Kernighamand Ritchie invents C(K&R C)
- 1980 **Bjarne Stroustrup creates** "C with classes"
- 1983 "C with classes" is renamed as "C++"
- 1989 The C standard is ratified (ANSI C, C89)
- 1995 The ANSI committee releases a draft of the C++ standard
- 1998 An official C++ standard is adopted(C++98)

- C++ influenced by C
  - Procedural programming
  - Object-oriented programming
  - Generic Programming



TIOBE index

순위	TIOBE Index (Aug. 2020)	Top programming languages on github (Apr. 2019)	The top programming languages by IEEE (2018)
1	C	JavaScript	Python
2	Java	Java	C++
3	Python	Python	Java
4	C++	PHP	C
5	C#	C++	C#
6	Visual basic .Net	C#	PHP
7	JavaScript	TypeScript	R
8	R		JavaScript
9	PHP		Go
10	SQL		Assembly

출처: TIOBE index, github, IEEE spectrum

PL Popularity

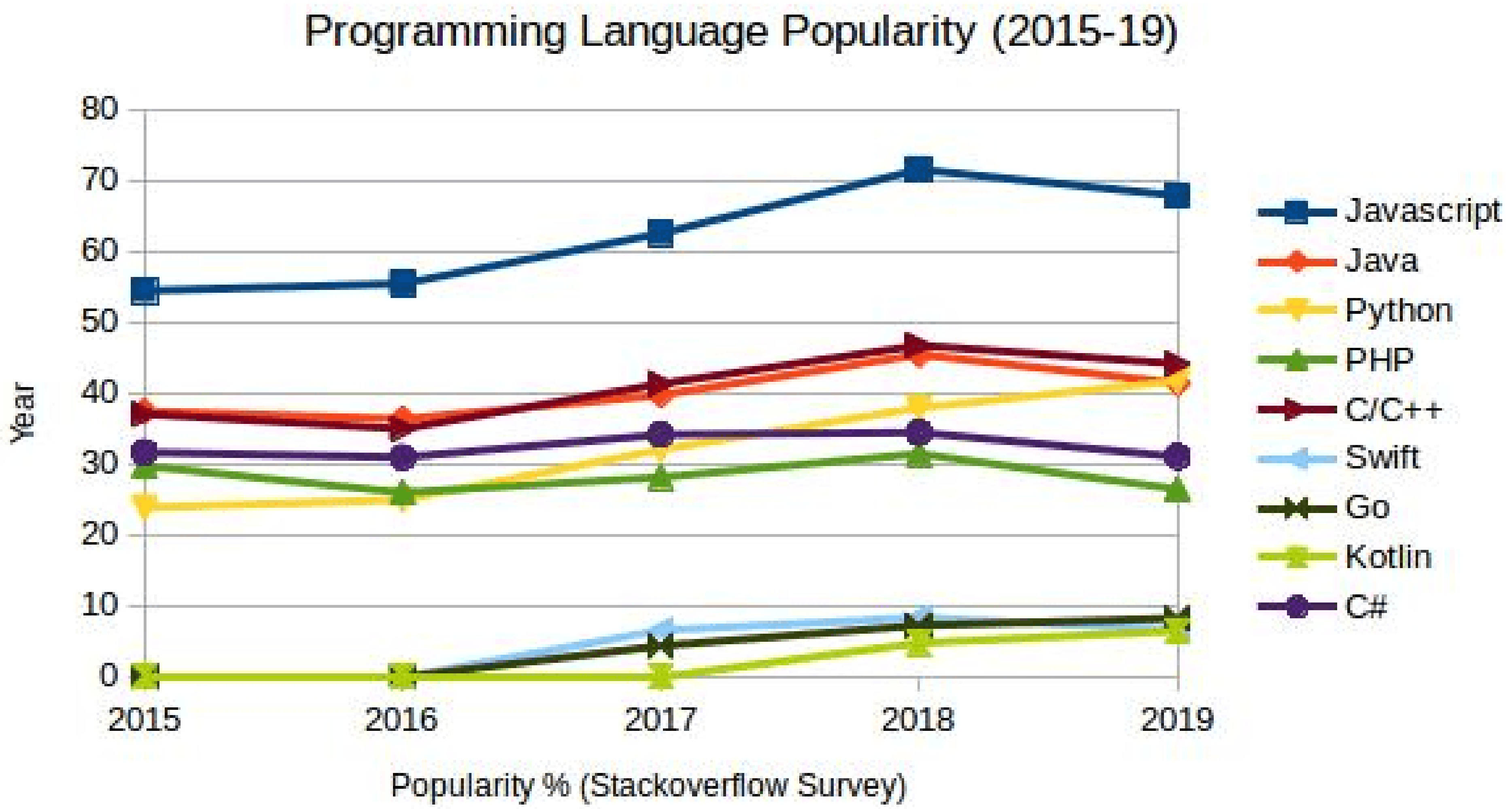
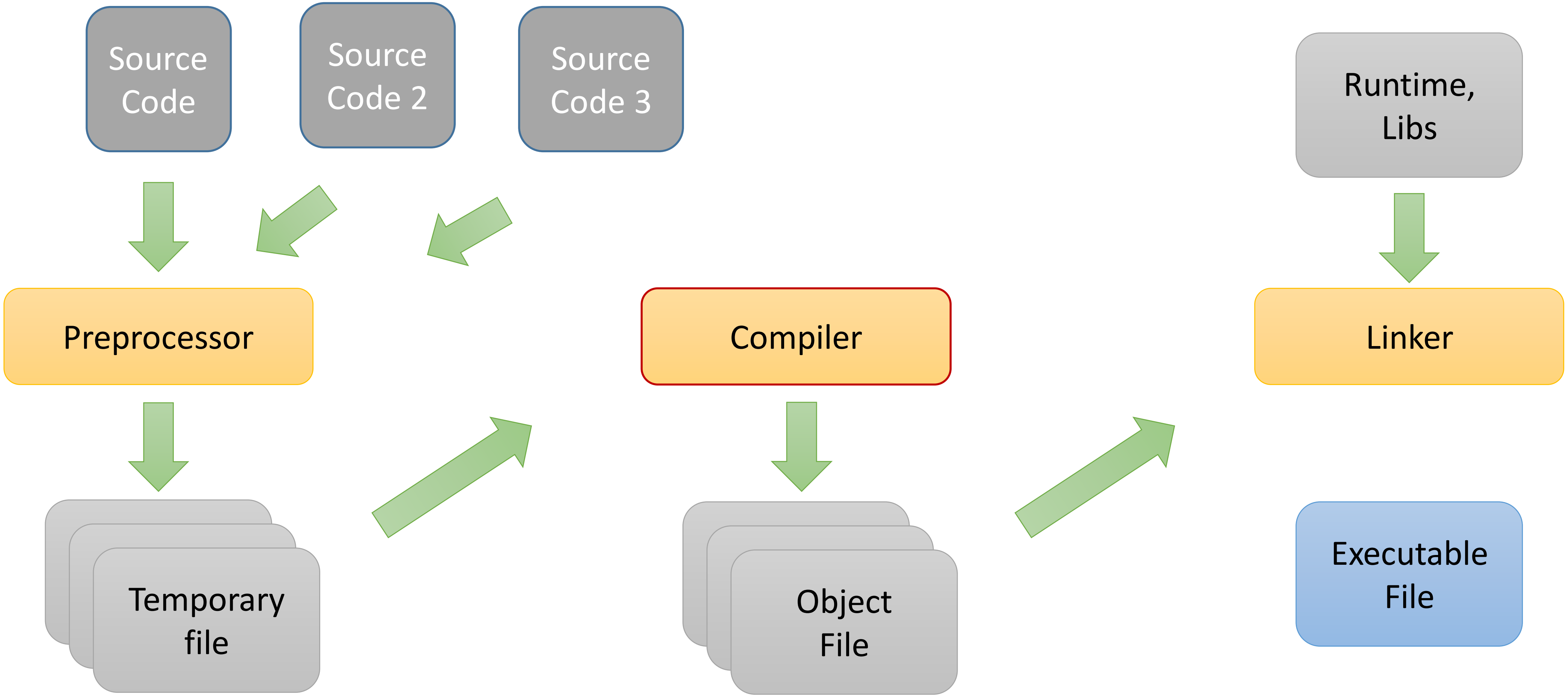


Image Source: codinginfinite

Why C++?

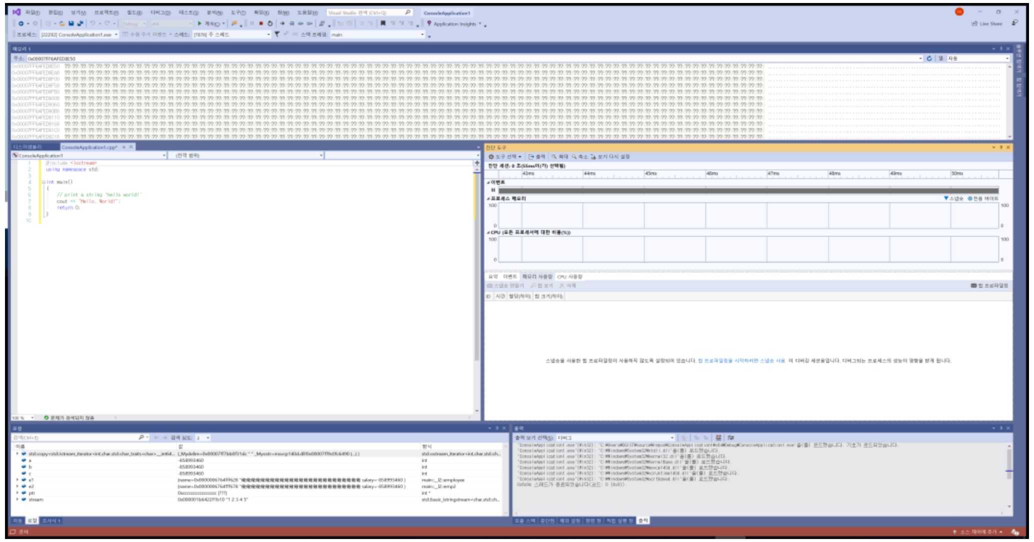
- Popularity
  - one of the largest code base
- Performance
  - the main development language for many system/ applications that require good performance
- Flexibility
  - low-level to high-level
- Productivity
  - libraries and development tool support

C++ Compiling Process



IDE (Integrated Development Environment)

- Ex.)
  - Windows : MS Visual Studio, Editplus, …
  - Linux : g++ + vim, notepad++, …
  - macOS : Xcode, …
  - (for this class, VS 2019 community version?)



```
#include <iostream>
using namespace std;

int main()
{
    // print a string "hello world!"
    cout << "Hello, World!";
    return 0;
}
```

예제: Hello World!

```
/* 예제 파일 */
#include <iostream>
using namespace std;
int main()
{
    int num1, num2;
    num1 = 1; num2 = 2;
    // print a string 'hello world!'
    std::cout << "Hello, World!" << endl;
    cout << num1 <<"+"<< num2 <<"="<< num1+num2;
    return 0;
}
```

```
Hello, World!
1+2=3
```

ANY QUESTIONS?