430.211 Programming Methodology (프로그래밍 방법론)

Intro & C++ Basics

Lab 1 Week 1

Fall 2025

Machine Intelligence & Data science LAB

Outline

- **Lab 1 Overview**

- Lab Introduction
  - Elice Guide
  - Assignment
  - Attendance Check

- C++ Basics

Machine
Intelligence &
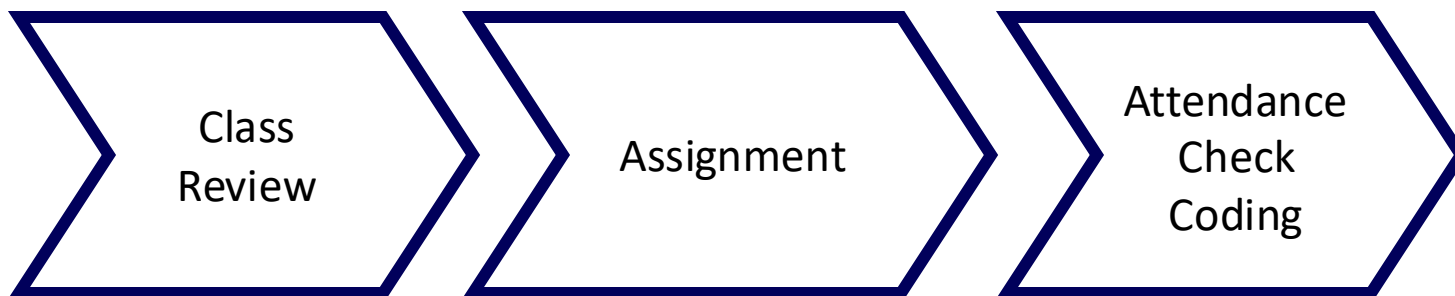Data science LAB

Lab 1 Overview

- TA Info (Lab 1 TA)
- Jungyoon Hwang (danieljyh@snu.ac.kr)
- Juhyeon Park (parkjh9229@naver.com)
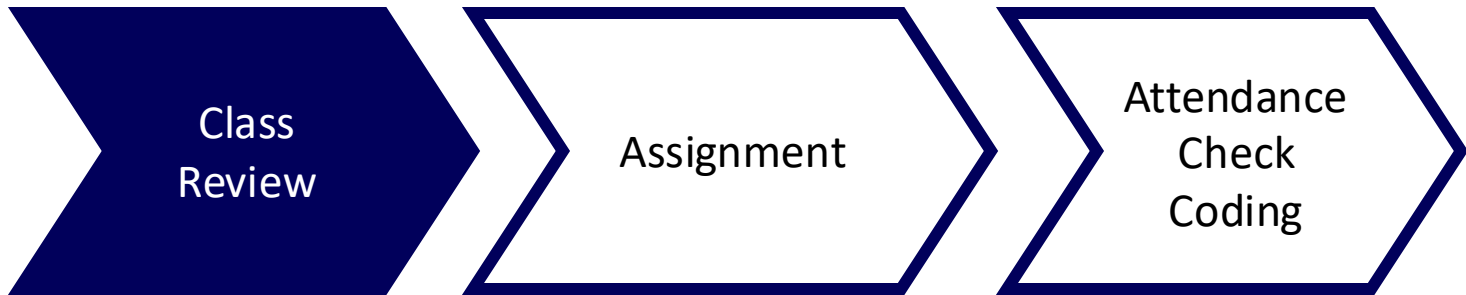- Yongho Kim (peterkim98@snu.ac.kr)

Lab 1 Overview

- Overview

```
Class        Assignment    Attendance
Review                     Check
                           Coding
```

- Q&A
• Public : ETL question board
• Private : ETL question board
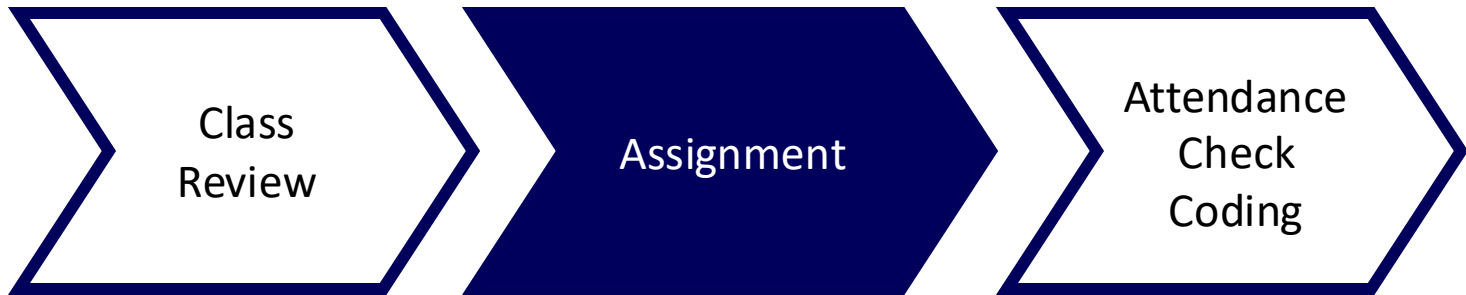   Lab1 → Yongho Kim (peterkim98@snu.ac.kr)

Lab 1 Overview

- Overview

- Lab 1 focuses on reviewing the class material

- Based on coding 👦👩‍💻

Lab 1 Overview

- Overview

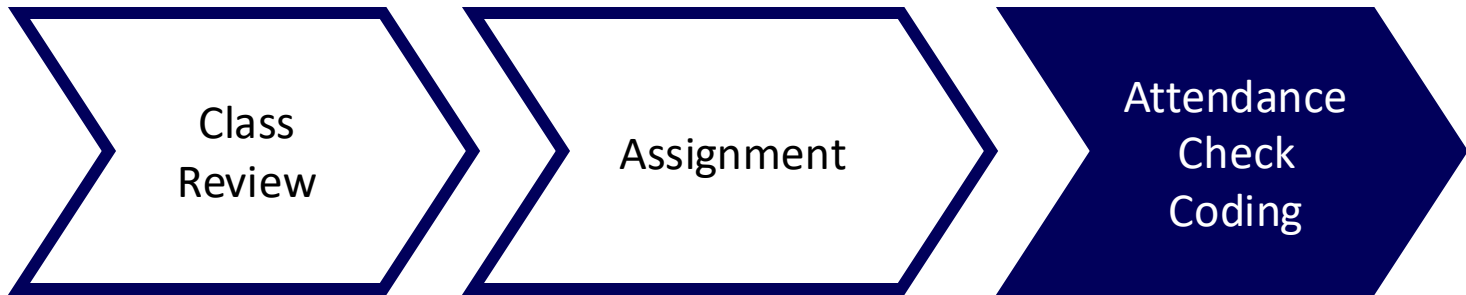| Class Review	Assignment	Attendance Check Coding

- Problems are related to the lecture of the week

- TA will give you a brief explanation

- Due: one hour before the next lab 1 session
  (~14:30 Friday)

Lab 1 Overview

- Overview

| Class Review | Assignment | Attendance Check Coding |

- Attendance will be checked with a simple programming problem
- After your solution is confirmed by TAs, you are free to leave.
- Problems will be revealed at the lab session

Outline

- Lab 1 Overview

- Lab Introduction
  - Elice Guide
  - Assignment
  - Attendance Check

- C++ Basics

Machine
Intelligence &
Data science LAB

Elice Guide

Elice Guide

Elice Guide

- Read the description carefully

- And fill in the TODO

- "실행" is not graded!

Elice Guide

- Press "제출" button to test your code

- You can resubmit as much as you need
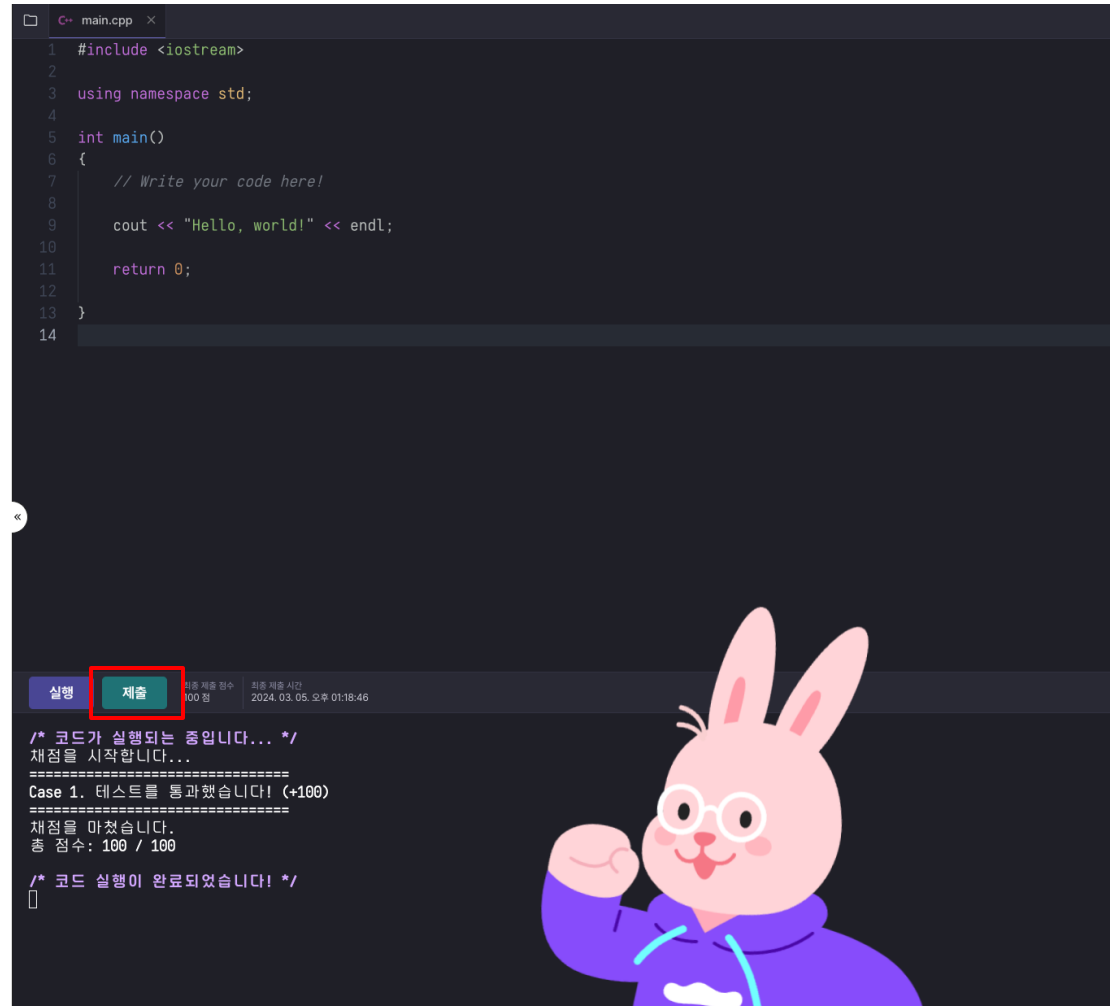
Lab Assignment

- Your assignment will be graded on the code on Elice
    - There will be additional hidden cases

- Due: one hour before the next lab 1 session
      (~14:30 Friday)

- After the deadline, the assignment tab will be closed

Attendance Check

- Your attendance will be checked with a simple problem

- After the class is over, raise your hand show your screen to TAs after pressing "제출" and getting 100% on Elice

- You are free to leave after your attendance is checked

Machine
Intelligence &
Data science LAB

Outline

- Lab 1 Overview

- Lab Introduction
  - Elice Guide
  - Assignment
  - Attendance Check

- C++ Basics

C++ Basic Structure

- Let's think of a simple code from python
  - Input a number
  - Print the inputted number

< Python >

```python
val = input("Enter a number?")

print("Your number", val)
```

Enter a number? :
5

Your number : 5

terminal

```
Enter a number? : 5

Your number : 5
```
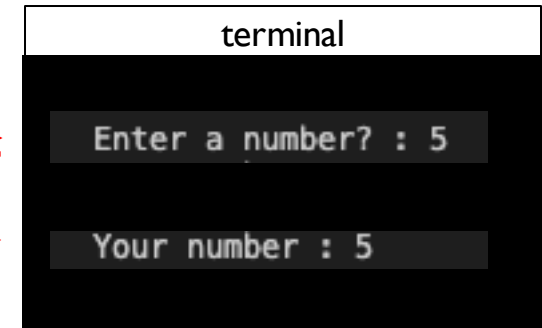
C++ Basic Structure

- Python built in functions

< Python >

```python
val = input("Enter a number?")

print("Your number", val)
```

Enter a number? :

5

Your number : 5

terminal

```
Enter a number? : 5

Your number : 5
```

- Input() and print() are built-in function to read input from the keyboard and to display output.
  - Input() returns as a string type (type cast)
  - Output() can display many complex types

https://www.geeksforgeeks.org/c-c-include-directive-with-examples/
http://www.qnx.co.uk/developers/docs/6.5.0/index.jsp?topic=%2Fcom.qnx.doc.dinkum_en_cpp%2Flib_cpp.html

C++ Basic Structure

- Ex 1. Console I/O and Namespace

```cpp
#include <iostream>


int cout(void){
    return 3;
}



int main(void){
    int val;

    std::cout << "Enter a number? : ";
    std::cin >> val;

    cout << "Your number : " << val << std::endl;

    return 0;
```

What happens?

Machine
Intelligence &
Data science LAB

C++ Basic Structure

- Ex 1. Console I/O and Namespace

```cpp
#include <iostream>


int cout(void){
    return 3;
}



int main(void){
    int val;

    std::cout << "Enter a number? : ";
    std::cin >> val;

    cout << "Your number : " << val << std::endl;

    return 0;
}
```

What happens?

Error: invalid operands to binary expression('int()' and 'constant char [15]')

C++ Basic Structure

- **Namespace**
  - Namespace, scope resolution operator(::) is covered later in Chap 11, Chap 6
  - When programs use libraries from a variety of sources, the possibility of *name collision* may exist.

Variables, Expressions

- Ex. 2 Program for total sum of your lift
    - 1. Use 3 variables (squat, dead, bench)
    - 2. Print the total sum of the 3 inputs

- Result

```
Squat? (Kg): 110
Deadlift? (Kg): 180
Benchpress? (Kg): 80
Your total lift is 370
```

Variables, Expressions

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

int main(void){

    int squat, dead, bench = 0;
    // int squat(0), dead(0), bench(0) initialization

    cout << "Squat? (Kg): ";
    cin >> squat;

    cout << "Deadlift? (Kg): ";
    cin >> dead;

    cout << "Benchpress? (Kg): ";
    cin >> bench;

    //print total weight
    cout << "Your total lift is " << squat + dead + bench << endl;

    return 0;

}
```

Machine Intelligence & Data science LAB

Variables, Expressions

- Ex. 3 Celsius to Fahrenheit conversion

  - Is the program running correctly?

```cpp
1    #include <iostream>
2    using namespace std;
3
4    int main(void){
5
6        double c, f = 0;
7
8        cout << "Enter current temperature (in Celsius): ";
9        cin >> c;
10       f = (9/5) * c + 32.0;
11       cout << "Current temperature in Fahrenheit: " << f << endl;
12
13       return 0;
14   }
```

Machine
Intelligence &
Data science LAB

Variables, Expressions

- Ex. 3 Celsius to Fahrenheit conversion

```cpp
#include <iostream>

using namespace std;

int main()
{

    double c, f = 0;

    cout << "Enter current temperature (in Celsius): ";
    cin >> c;
    f = (9/5) * c + 32.0;
    /* Try the following:
    1) ((double)9/5)      O
    2) (9/(double)5)      O
    3) (double)(9/5)      X
    4) (9.0/5)            O
    */
    cout << "Current temperature in Fahrenheit: " << f << endl;

    return 0;
}
```

Machine
Intelligence &
Data science LAB

Variables, Expressions

- Ex. 4 Celsius to Fahrenheit conversion (+ precision)
  - 1. Use
    ```
    cout.setf(ios::fixed);
    cout.setf(ios::showpoint);
    cout.precision(2);
    ```

  - 2. Add additional variable 'precision'

- Result

```
Enter current temperature (in Celsius): 24.88
Precision? : 1
Current temperature in Fahrenhiet: 76.784
Current temperature in Fahrenhiet (after precision): 76.8
```

Variables, Expressions

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

int main(void){

    double precision, c, f = 0;

    cout << "Enter current temperature (in Celsius): ";
    cin >> c;
    f = (9.0/5) * c + 32.0;

    cout << "Precision? : ";
    cin >> precision;

    cout << "Current temperature in Fahrenheit: " << f << endl;

    cout.setf(ios::fixed);
    cout.setf(ios::showpoint);
    cout.precision(precision);

    cout << "Current temperature in Fahrenheit (after precision): " << f << endl;

    return 0;
}
```

Machine
Intelligence &
Data science LAB

Variables, Expressions

- Ex. 5 Program for total sum of you lift (upgraded)
    - 1. Use constant variable for pound (1kg = 2.20462 lbs)
    - 2. Do not add additional variables other than precision
    - 3. Round to 3rd decimal place

- Result

```
Squat? (Kg): 70
Deadlift? (Kg): 100
Benchpress? (Kg): 60
Your total lift (Lb) is 507.063
```

Variables, Expressions

```cpp
1    #include <iostream>
2    using namespace std;
3
4    const double pound = 2.20462;
5
6    int main(void){
7
8        int squat, dead, bench = 0;
9        // int squat(0), dead(0), bench(0) initialization
10
11       cout << "Squat? (Kg): ";
12       cin >> squat;
13
14       cout << "Deadlift? (Kg): ";
15       cin >> dead;
16
17       cout << "Benchpress? (Kg): ";
18       cin >> bench;
19
20
21       cout.setf(ios::fixed);
22       cout.setf(ios::showpoint);
23       cout.precision(3);
24       //print total weight
25       cout << "Your total lift (Lb) is " << (squat + dead + bench)*pound << endl;
26
27       return 0;
28
29   }
```

Machine
Intelligence &
Data science LAB

Operators

- Ex. 6 Increment examples
  - What would be the output?

```cpp
#include <iostream>

using namespace std;

int main()
{
    int i(1), j(2);

    cout << i++ << endl;

    cout << ++i << endl;

    cout << --j << endl;

    cout << j-- << endl;

    return 0;
}
```

Attendance Check Coding

- ## Hour / Minute / Second

  Write a program that:

  - Inputs an integer that represents a length of time in second.
  - Outputs the converted hour, minute, second from the input
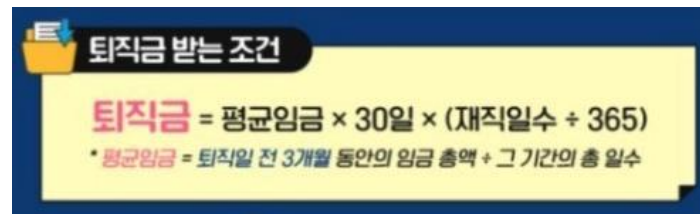   (ex) 50391 sec -> 13hr 59min 51sec

  - **Result**

  ```
  /* 코드가 실행되는 중입니다... */
  Total second? : 50391
  13hr 59min 51sec

  /* 코드 실행이 완료되었습니다! */
  ```

Assignment

- Severance Pay calculator
  - Write a program that calculates severance payment.
  - Here, we use the formula below for its calculation.

$$Severance = Average\ wage\ \times\ 30\ \times\ (\#\ of\ working\ days\ \div 365)$$

  - The number of working days(재직일수) is given as an input

  (We assume every employee has worked at least a year, 재직일수 ≥ 365)

  - Use constant variable for average wage(평균임금) (avgWage=60,000)

  - Round to the nearest thousands (100의 자리에서 반올림).

Submission

- Attendance Check Code
  - After this week, raise your hand after you are finished
  - **There will be no attendance checks this week**

- Assignment
  - Write your code on Elice and submit
  - **This week's assignment will not be graded**

- Lab2: Sign and submit Code of Ethics on Elice (KOR or ENG)

Thank you!