Overview

Modern cpp Programming lecture 1

Why cplusplus?

- Why we learn programming languages??

  - To write program

  - To learn computational logic

  - To automate certain job

  - To obtain good job

  - for fun...

Why cpp?

- Okay. Then why cpp?

- Let's think about "human-language".

- Someone is learning *Spanish Language*...Why?

  – To visit Spain

  – To visit Latin America (Mexico, Brazil, Chile…)

  – To work with Spanish

  – To study Spanish culture.

- Nobody will learn Spanish to visit China!!

Why cpp?

- Each human language has different purpose to learn

- Why programming languages don't?

- Each programming language has different purpose to use

  – Python: exploratory programming, data processing

  – C: Embedded programming, developing system-level program

  – Javascript: front-end programming, server-side programming

  – Java: server-side programming, mobile applications (Android!!)

  – C#: Developing game (Unity), windows applications (.NET)

  – And others…

Why cpp?

- Main purpose of cpp

  - When we need "high performance"

    - Game

    - Embedded programming

    - System-level programming

    - Core of machine learning libraries

  - Okay, cpp is fast and efficient...why?

    - To answer the question, let's briefly study the background of cpp

Cpp

- Released in 1983

- Designed / Developed by *Bjarne Stroustrup*

- Initially called "C with Classes"

- Main requirements:

  – Class, Inheritance, Polymorphism

  – Inline expansion

  – template, operator overloading...

  – we will learn!!!!

Cpp is difficult

- Old-fashioned

  – released in 1983

  – compared to other modern languages…

    • Python(1981), Java(1995), Go(2012), Swift(2015)…

  – hard to use

    • No modern programming features

    • Complex

    • High LOC (lines of code)

    • pointer hazard

Cpp is difficult

- filter out even numbers from oldArray!!

  - cpp
    ```
    vector[int] newArray;
    for(int i = 0; i < arrayLength; i++) {
        if(oldArray[i] % 2 == 0) {
            newArray.push(oldArray[i]);
        }
    }                                          too long...
    ```

  - javascript
    ```
    let newArray = oldArray.filter(e => (e % 2 == 0));
    ```
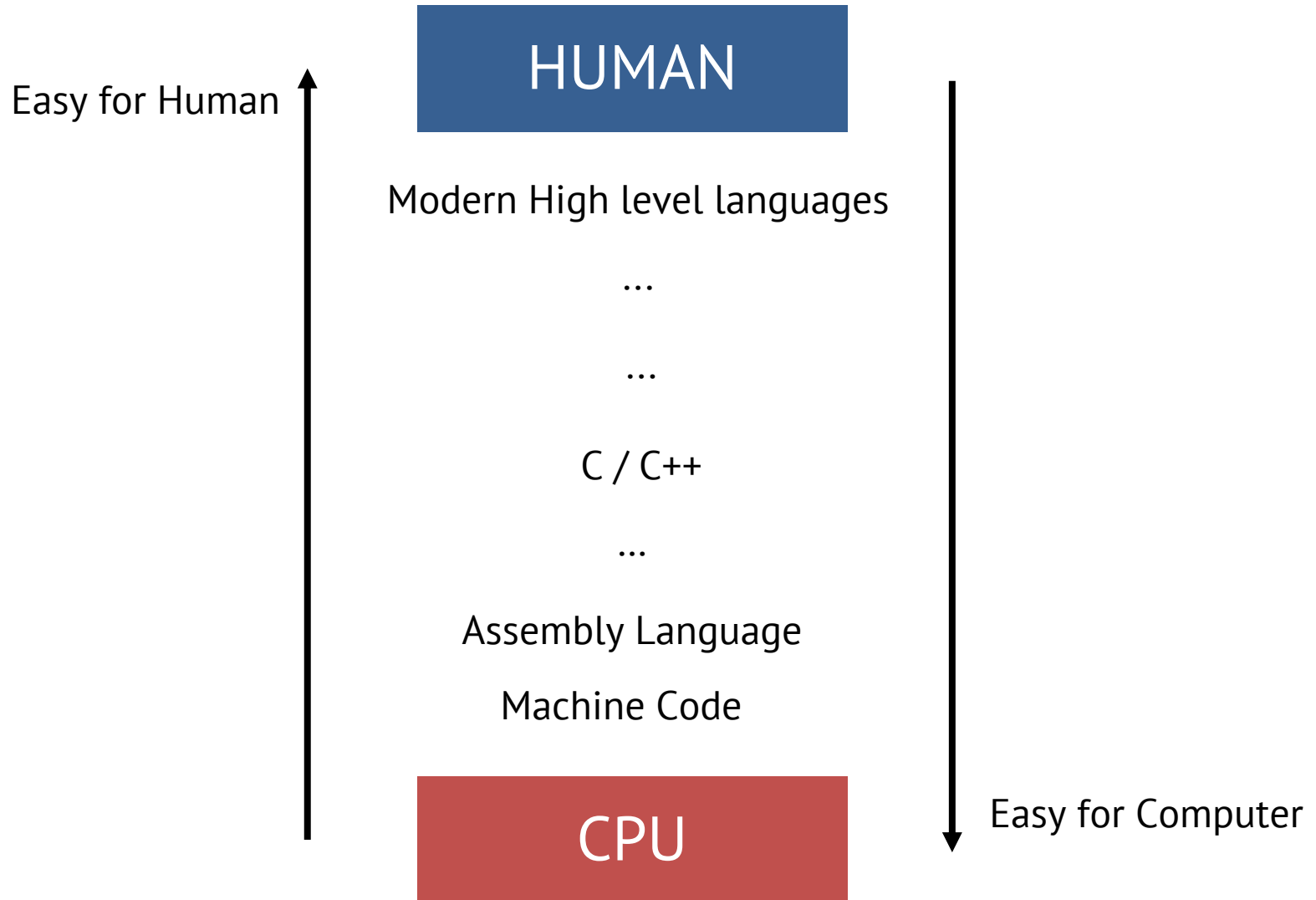
  - ocaml
    ```
    List.filter (i mod 2 = 0) oldArray
    ```

Cpp is difficult

- Computer-friendly

- You might heard that computers can only understand the language consists with 0 and 1.

- The language is called *Machine Code*

  - Cpp i*s* really close to Machine Code

  - This makes cpp hard (not human-friendly)

  - **Paradoxically, this is why we should learn cpp!!**

Cpp is difficult

Cpp is difficult

- Suitable to understand *computer system*

- Good to understand the fundamentals of programming

- Also...

  - Old == famous!! lots of reference

  - good to understand OOP

  - well-designed / defined libraries

  - Various useful *basic* features

    - After you finished cpp, no language will bother you

In this course...

- Basic features of *programming*

  – Already finished!! (Prerequisites)

- Basic features of *cpp*

  – pointer / reference

  – dynamic allocation

  – exception

  – template

In this course...

- C++ Standard Library

  - Useful tool for programming!!

    - Container : array, queue, map, vector...

    - Algorithm

    - String

    - input / output stream

    - Numeric

    - C Standard Libraies

      - math, time, stdlib...

In this course...

- Object-Oriented Programming

  - You're developing the program:

    - which describes the interaction b/w people

    - or even the interaction b/w people and country

    - better to design *something* which represents a person and country

    - *something* is a **Object**

  - One of programming paradigms

    - procedural programming

    - Object-Oriented Programming

    - Functional programming

In this course…

- Object-Oriented Programming

  - Inheritance

    - You made *object* which represents **human**

    - Maybe you can use **human** object designing **Alice, Bob** objects.

  - Polymorphism

    - Okay…You now have **Alice, Bob, Carol, Dave, Eve**…

    - You need to design some logic to make them sing a song

    - which is better?

      - function sing(Alice), sing(Bob), sing(Carol)…
      - function sing(Human) and substitute Alice~Eve

    - Quite evident

In this course...

- Object-Oriented Progamming

  - Information hiding / Encapsulation

    - now you can make people sing

    - however, you don't need to reveal vocal cords!!

    - only you need is to reveal the method to make people sing

    - capsulate *Human*

In this course...

- Useful skills

  - GUI Programming

    - makes *real* application which people can use

    - will use **Qt**

  - Meta Programming

    - powerful feature of cpp!!

    - will use *template*

      ```
      vector<int> numArray;
      vector<string> stringArray;
      ```

In this course…

- Optional
  - Multi-thread programming
    - a.k.a parallel programming
    - In nowadays, computers have more than 4~6 cpu cores
    - To make efficient program, you should use all of them
  - Polyglot programming
    - using multiple languages at once!!
  - Design Pattern
    - how to organize programming features??
    - What is good design??

After the course, you will …

- be able to develop something you want using cpp

- understand the basics of computer

- be ready to learn other programming languages

- be ready to learn other fields!!

  – Machine learning

  – Web / App development

  – Data crawling / processing

  – So many things to do!!

- be familiar to use / manage computer

- challenge!!

  – integrate computer with your domain knowledge to solve problem

  – Competition, Hackathon…

Thank you!!

contact: jeonhyun97@postech.ac.kr