



INTRODUCTION

PROGRAMMING TECHNIQUES

ADVISOR: Trương Toàn Thịnh

CONTENTS

- Goals & contents
- Exercises

GOALS & CONTENTS

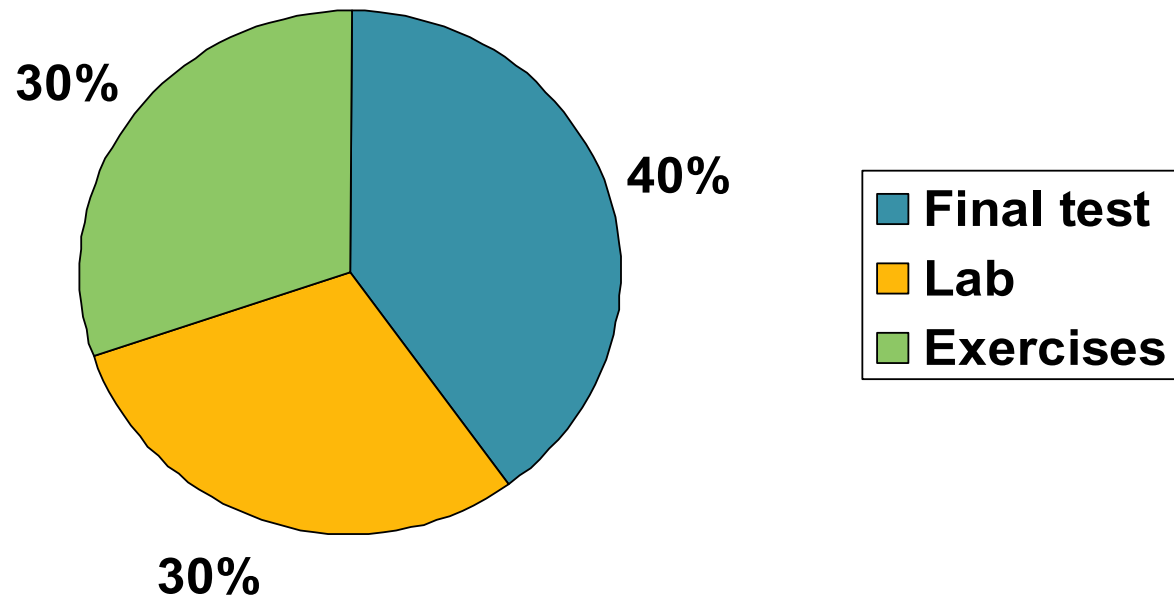
- Review all basic concepts of programming languages, such as C/C++
- Apply C/C++'s techniques to solve the problems
- Understand some basic and advanced programming techniques
- Programming environment:
 - Visual C++.NET 2013 to 2019.

GOALS & CONTENTS

- Train some software implementation skills in reality
- Organize source-code and build the highly-reuse package
- Achieve some background knowledge of basic data structure
- Understand the basic knowledge of programming science
- Know some basic algorithms

GOALS & CONTENTS

- Grade-scale



GOALS & CONTENTS

- Class

- Organize a group of ... members:

- Discuss

- Do exercises

- Student: read more material

- Discussion:

- Forum:

<https://courses.fit.hcmus.edu.vn/ctdb/login/index.php>

- Advisor: Trương Toàn Thịnh

- Email: ttthinh@fit.hcmus.edu.vn.

- Room: number I82, software engineering department, Information technology faculty, VNU-HCM University of Science.

GOALS & CONTENTS

- References:

- **Kỹ thuật lập trình C** – Prof Phạm Văn Ất
- **Kỹ thuật lập trình** – Prof Trần Đan Thư, PhD Nguyễn Thanh Phương, PhD Đinh Bá Tiến, Prof Trần Minh Triết, and MSc Đặng Bình Phương.
- **The C++ Programming Language** – Bjarne Stroustrup
- **Introduction to Algorithms** – Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein

CONTENTS

- Main contents
 - Using pointer: allocate, free...
 - Linked list data structure
 - Stack and queue
 - String processing
 - Search and Sort algorithms
 - Recursion
 - Dynamic programming
 - Pointer function & customized source code

CONTENTS

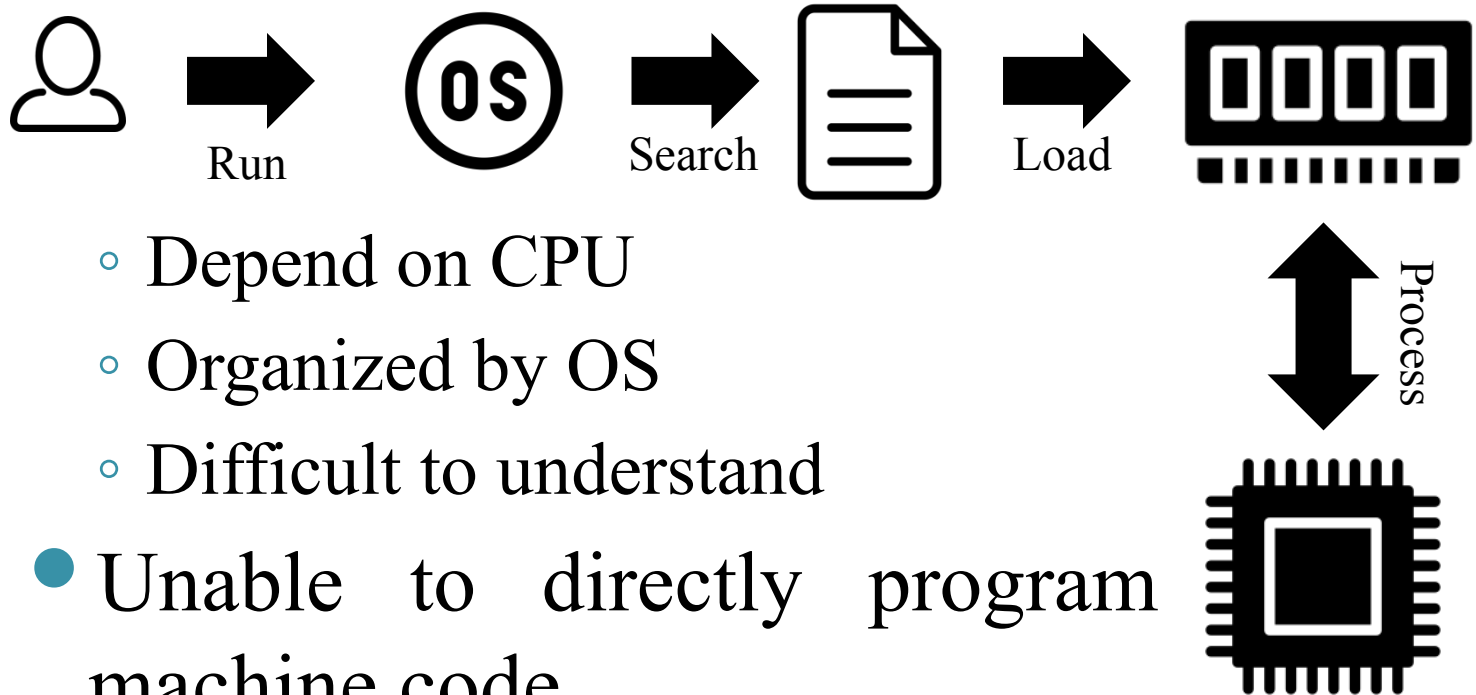
- Goals & contents
- Exercises

EXERCISES (REVIEWS)

- What program is
 - A sequence of instructions
 - To solve practical/academic problems
- Who programs is developer
- There are two basic programs
 - Machine code
 - Assembly & assembler

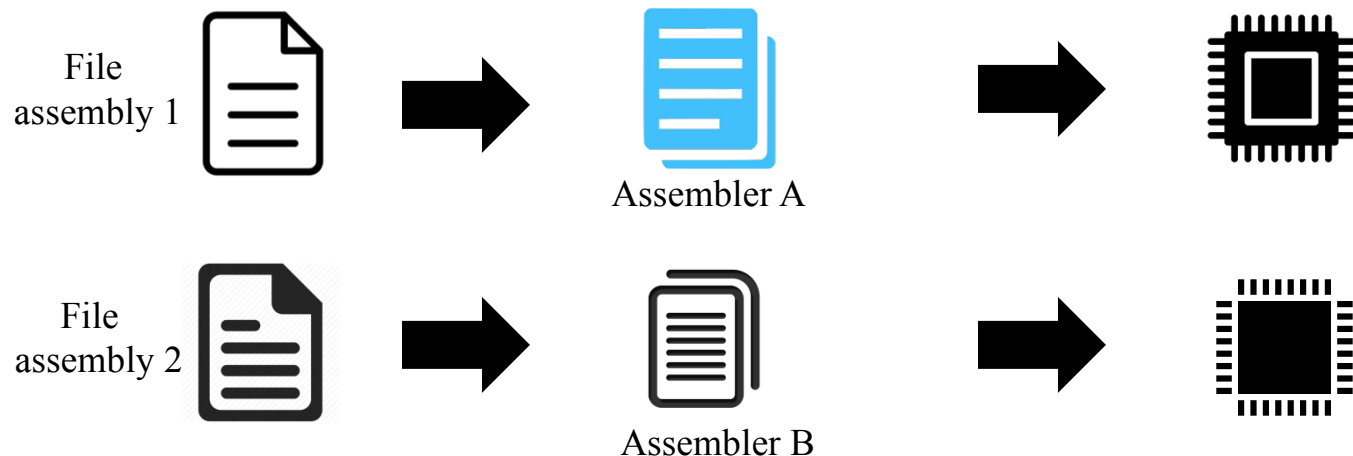
EXERCISES (REVIEWS)

- Process of machine code
- Some limitations



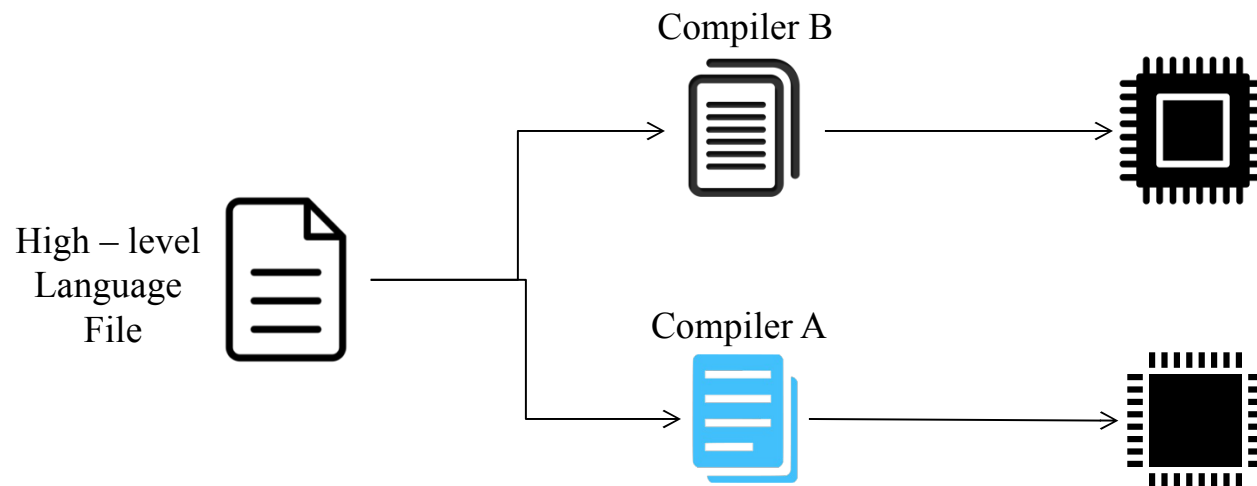
EXERCISES (REVIEWS)

- Assembly is a low-level language
 - Easier to understand than machine code
 - Need **Assembler** to translate into machine code
- Defect: depend on CPU
- Merit: take advantage of CPU



EXERCISES (REVIEWS)

- Traditional high – level programming language
 - More abstract than assembly
 - Easier to understand than assembly
 - More compatible than assembly

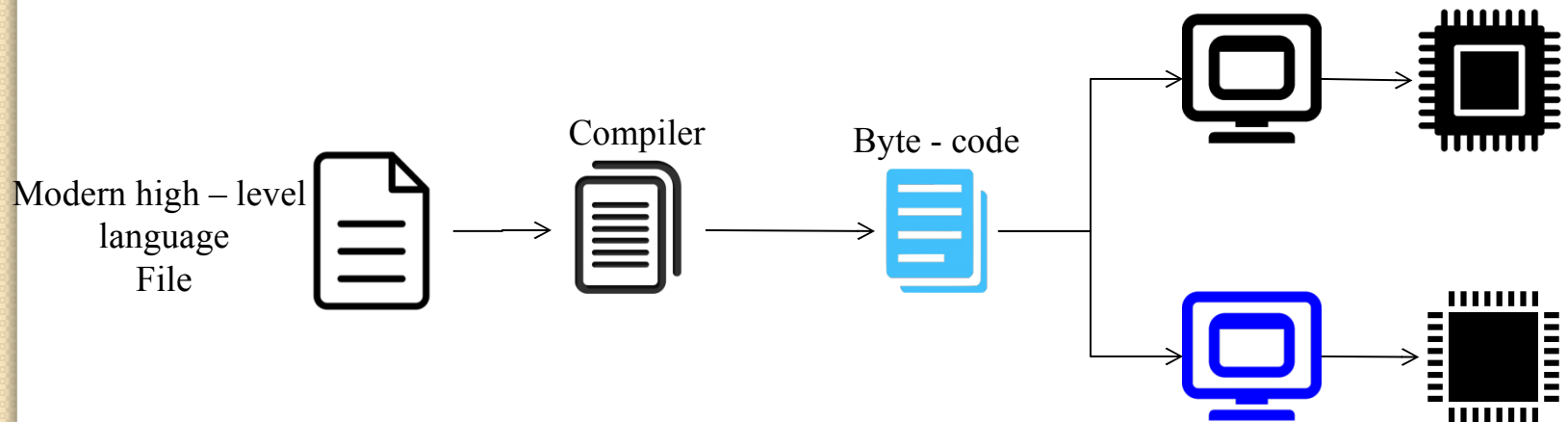


EXERCISES (REVIEWS)

- There are many languages with various objectives
 - Web applications: PHP, ASP.net, Ruby
 - Administration systems: Java, C#
 - Science computations: Fortran
 - Machine learning algorithms: Python
 - Operating systems: C/C++

EXERCISES (REVIEWS)

- Limitation of traditional high-level programming languages: their compilers directly translate into machine code
- The modern high-level programming languages' compilers translate into intermediate code (byte-code)



EXERCISES (REVIEWS)

- There are two high – level ones
 - Compiler
 - ▮ All source code is performed syntax analysis
 - ▮ Translates a programming language into another language (machine code + management information)
 - ▮ Results saved on hard-disk are execution files
 - Interpreter
 - ▮ Checks syntax at each line of codes
 - ▮ Perform its behavior directly

EXERCISES (REVIEWS)

- Process of compiling high-level language
 - Developer chooses another text-editor to compose source – code
 - Compiler translates it into target language (intermedia code)
 - Linker connects all intermedia codes into executable files
 - Run the program

EXERCISES (REVIEWS)

● Diagram

