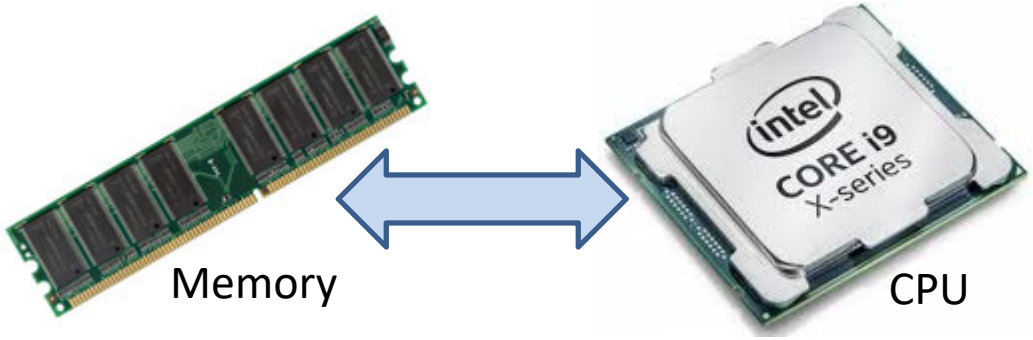
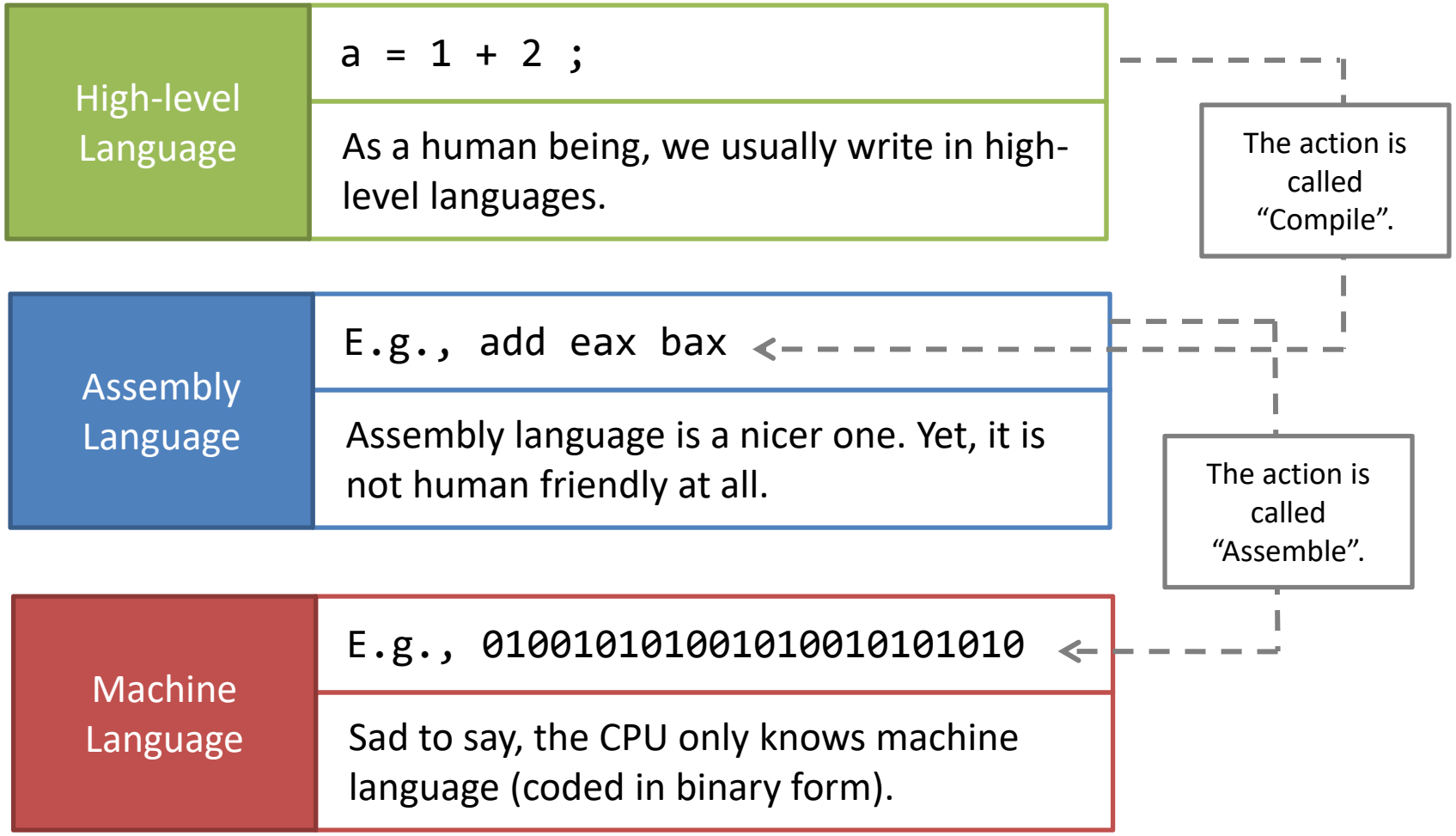
Introduction to Programming

What is Programming?

- **Computer** – a **machine** that manipulates **data (memory)** according to a set of **instructions (CPU)**
  - e.g.: Desktop PC, phone, PS5, smartwatch, car, …
- **Computer Program** – A set of instructions that tells the computer how to process the data
- **Programming** – Writing a computer program in a certain *language* to tell the computer what to do



Programming Languages

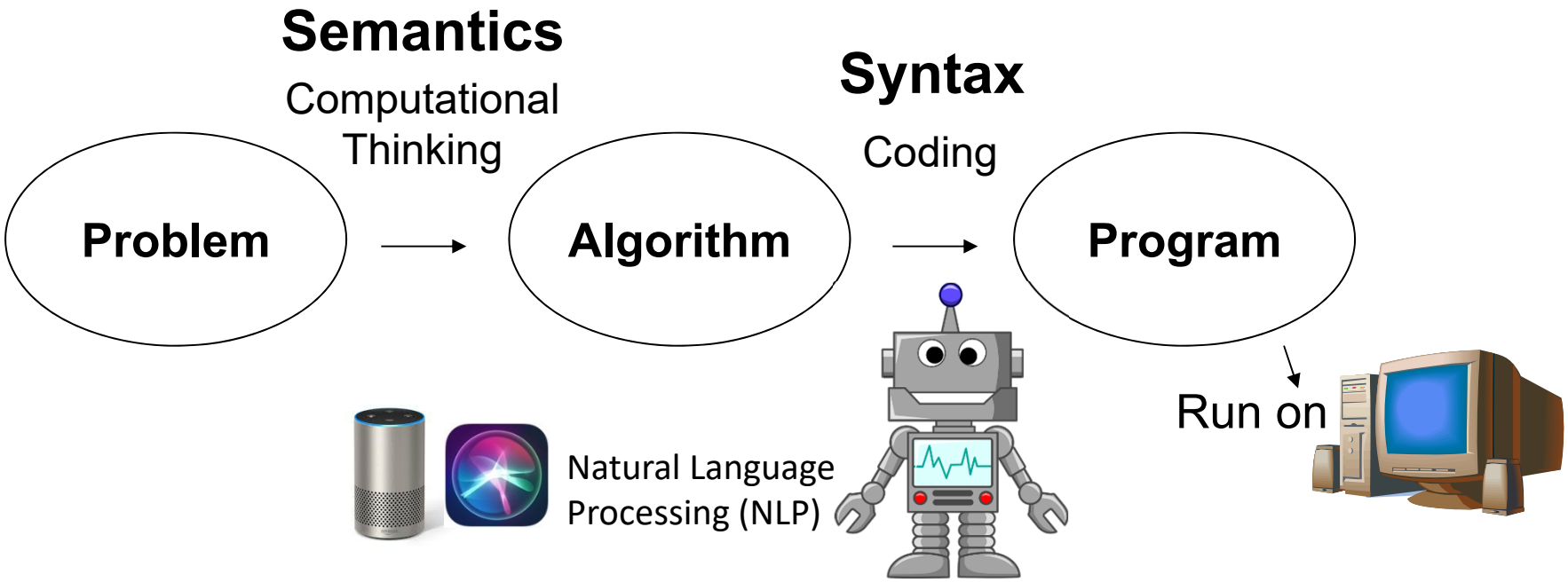


What is Problem Solving by Programming?

- It is a **process** that involves:
  - Understanding of how computers work
  - Analysis and modeling of the problem and the data
  - Acquire logic and procedural design concepts
  - Formulate an algorithm/procedure that one can translate into a program for computer to run
  - Automation
  - Efficiency: Time and Memory
  - Validation: Testing!!!
  - …

Also known as **Computational thinking!!!!!!**

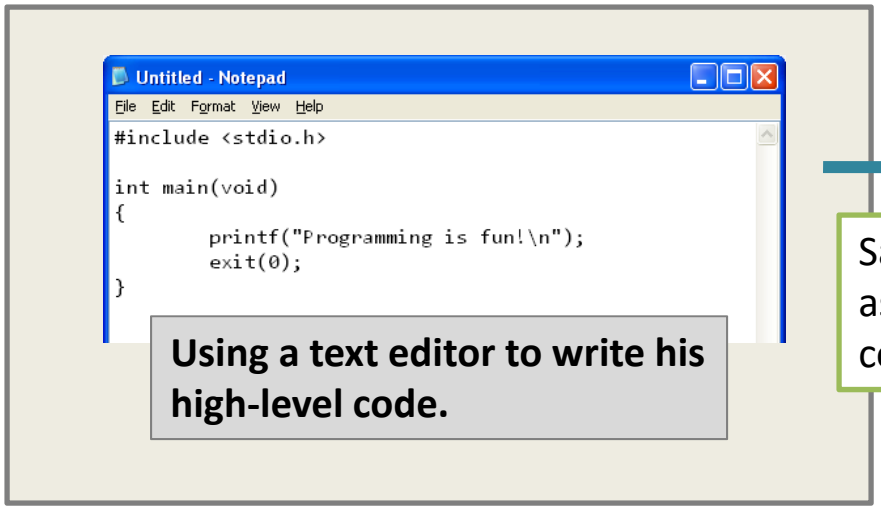
Programming Cycle



– **Computational thinking** is a mental process aiming at solving a problem by formulating a procedure/method that can be programmed and run on a computer

A recent project in 2021: https://openai.com/blog/openai-codex/
https://spectrum.ieee.org/programming-by-voice-may-be-the-next-frontier-in-software-development

Programming Cycle



Save his work as a program code file.

Compiler + Assembler

Machine code "MyProgram.exe"

Execution

Is the output correct?

No…then, do it again, i.e., debugging.

Yes!

* For every possible input scenario!!!

NOTE: A program is correct, only if it performs correctly for every possible input

Fundamentals of Programming

- Understand the basic concepts and principles of programming
- Understand and memorize the syntax of a programming language
- Turn your idea/solution into equivalent instructions
- Know what APIs (Application Programming Interface) are available and learn how to use them

Fundamentals of Programming (Explained)

- **Understand the basic concepts and principles of programming**
  - e.g.: variables, data type, functions, parameters, control structures, arrays, class, object, etc.
  - These concepts and principles are "universal".

- **Understand and memorize the syntax of a programming language**
  - Different languages have different syntax and grammar.
  - A program with syntax errors won't execute correctly.

- **Turn your idea and solution into equivalent instructions**
  - Analogous to learning English/Chinese – in addition to knowing the grammars and vocabularies, you also need to learn how to make sentences and compose essays.
  - Practices make perfect

- **Know what APIs (Application Programming Interface) are available and learn how to use them**
  - APIs here refer to collections of "codes" (written by other people) that offer programmers some commonly needed functions.
    - e.g., print data to screen, send data over network, display a window, etc.
  - If you can find the appropriate APIs to use in your program, you can potentially save a lot of time from writing everything from scratch.

Programming Tools

- Visual Studio
  - An Integrated Development Environment (IDE)
    - Editor + Compiler + Debugger + Other development tool
  - VS Community 2022 is free to download
    - https://visualstudio.microsoft.com/downloads/
- Code::Blocks
  - http://www.codeblocks.org/downloads/binaries (Free!)
  - See “00. UsingCodeBlocks.pdf” on Blackboard.
- Others: gcc on Linux, Xcode on Mac, etc.

Note: you may use whatever IDE for code development but please ensure your code still works on “codeSubmit”.