Lecture 1

Principles of Programming Language
and a bit of History

Do you have any idea what was the plan of Microsoft to sell? (Software)

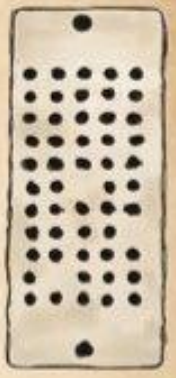
ADA LOVELACE

FIRST COMPUTER PROGRAMMER

The Analytical Engine

Lovelace's program turned a complex formula into simple calculations that could be encoded on punched cards and fed into Charles Babbage's Analytical Engine, a mechanical computer that he designed but never built. She published it in 1843, a century before the modern computer age.

"I want to put in something about Bernoulli's Number, in one of my Notes, as an example of how an explicit function may be worked out by the engine, without having been worked out by human head and hands first."



$$\frac{x}{e^x - 1} = \frac{1}{1 + \frac{x}{2} + \frac{x^2}{2 \cdot 3} + \frac{x^3}{2 \cdot 3 \cdot 4} + \&c.}$$

A Universal Computer

Lovelace did more than write the first computer program. She was also the first person to realise that a general purpose computer could do anything, given the right data and instructions.

"The Analytical Engine weaves algebraic patterns just as the Jacquard loom weaves flowers and leaves."

"Supposing, for instance, that the fundamental relations of pitched sounds in the science of harmony and of musical composition were susceptible of such expression and adaptations, the engine might compose elaborate and scientific pieces of music of any degree of complexity or extent."

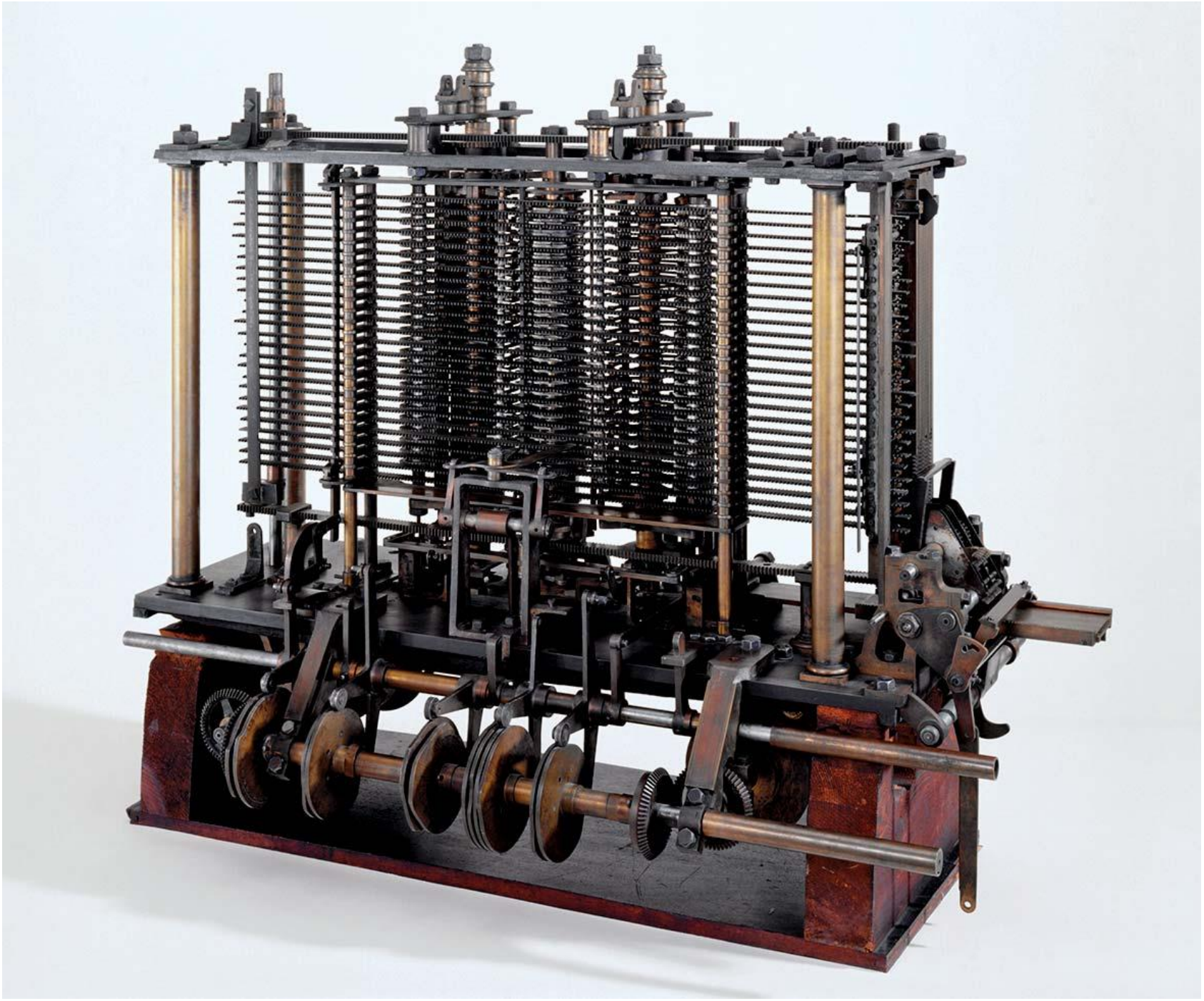
Augusta Ada King,
Countess of Lovelace
Born: 10 December 1815
Died: 27 November 1852

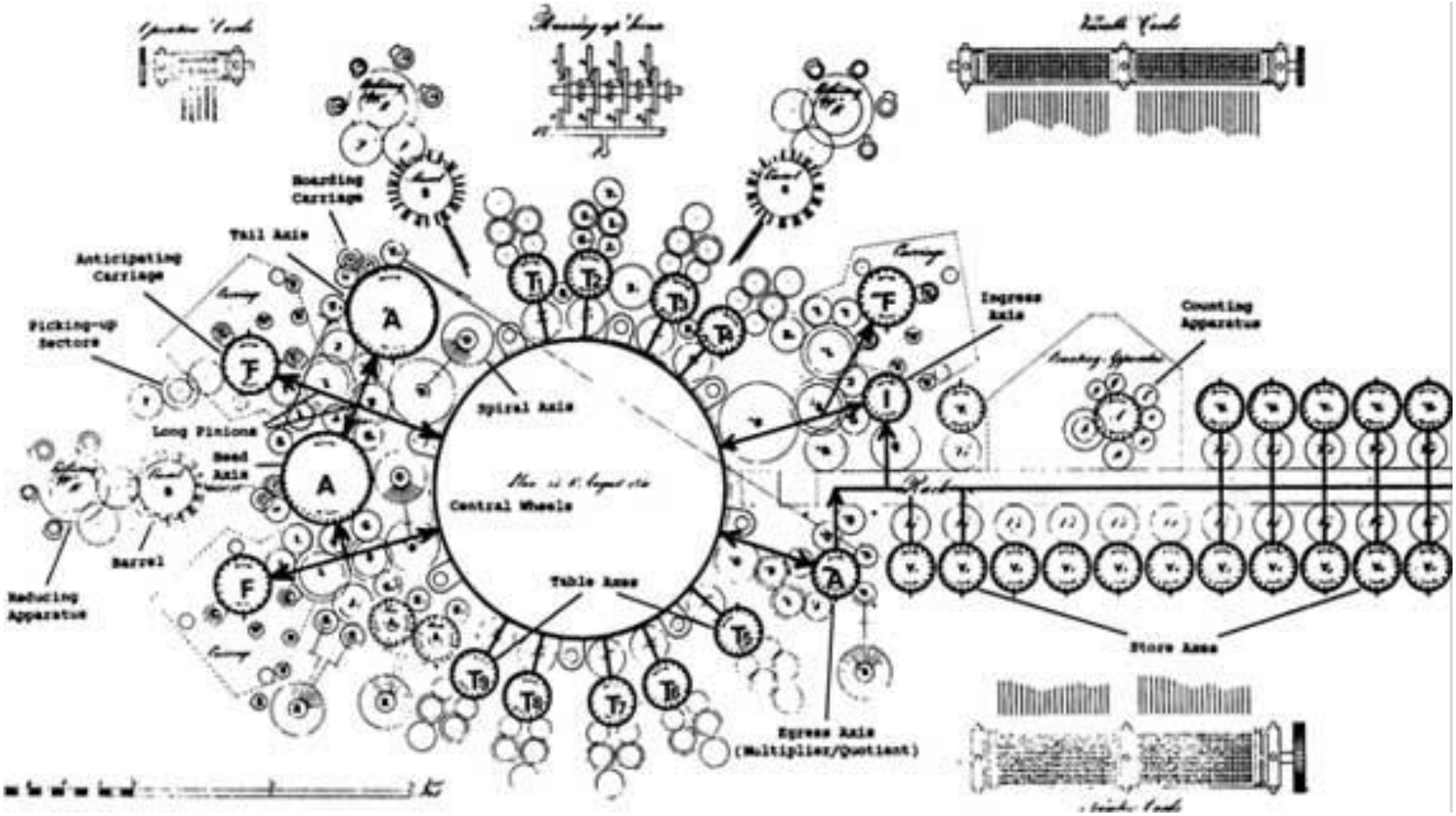
Alan Turing

Artifiicial Intelligence

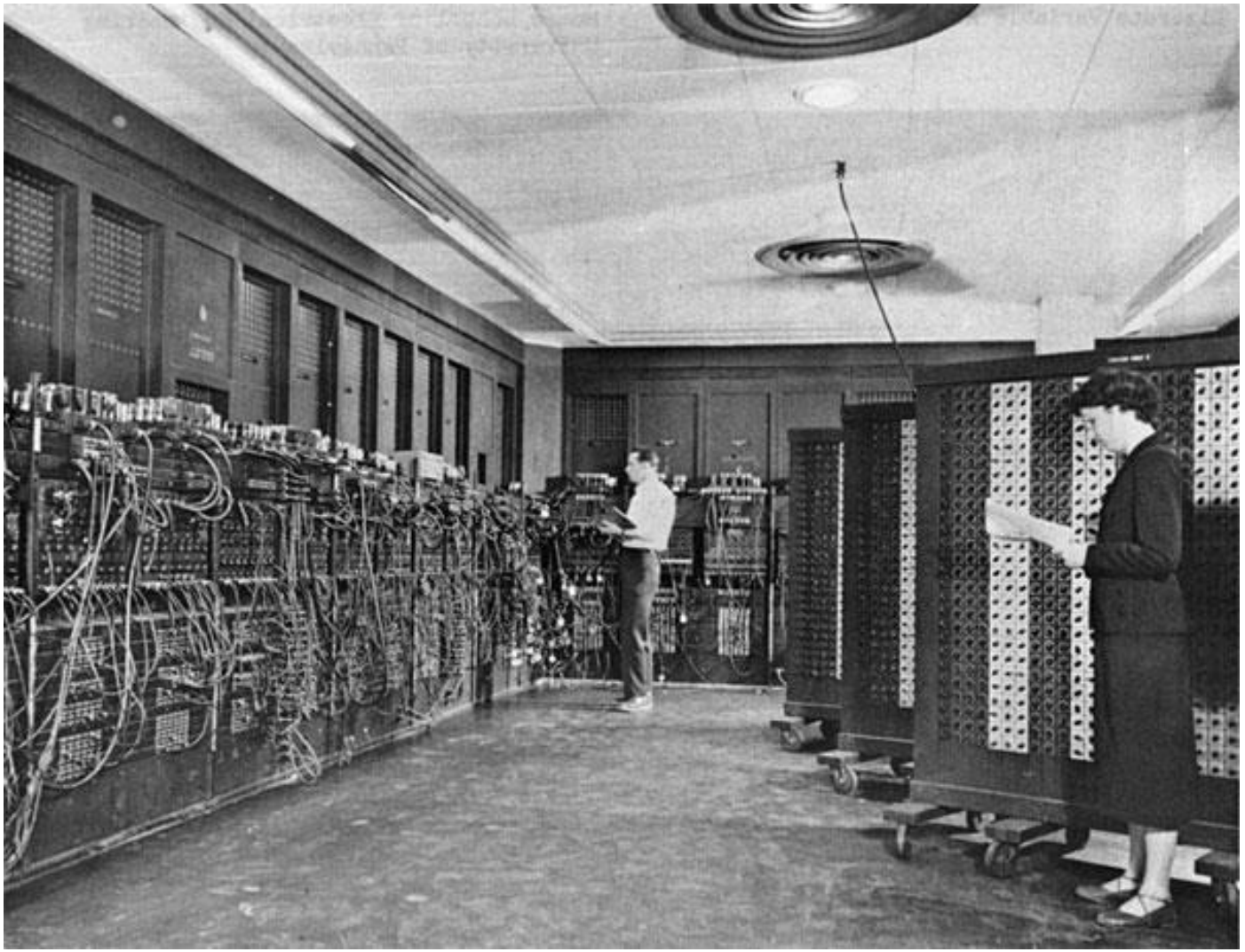
What is a Programming Language?

- Lets describe first natural language like English, Filipino etc.
  - Person to person
- In programming language:
  - Person to computer
- Definition by experts:
  - A programming language is a system for describing computation.
  - A system of signs used by a person to communicate a task/algorithm to a computer, causing the task to be performed.

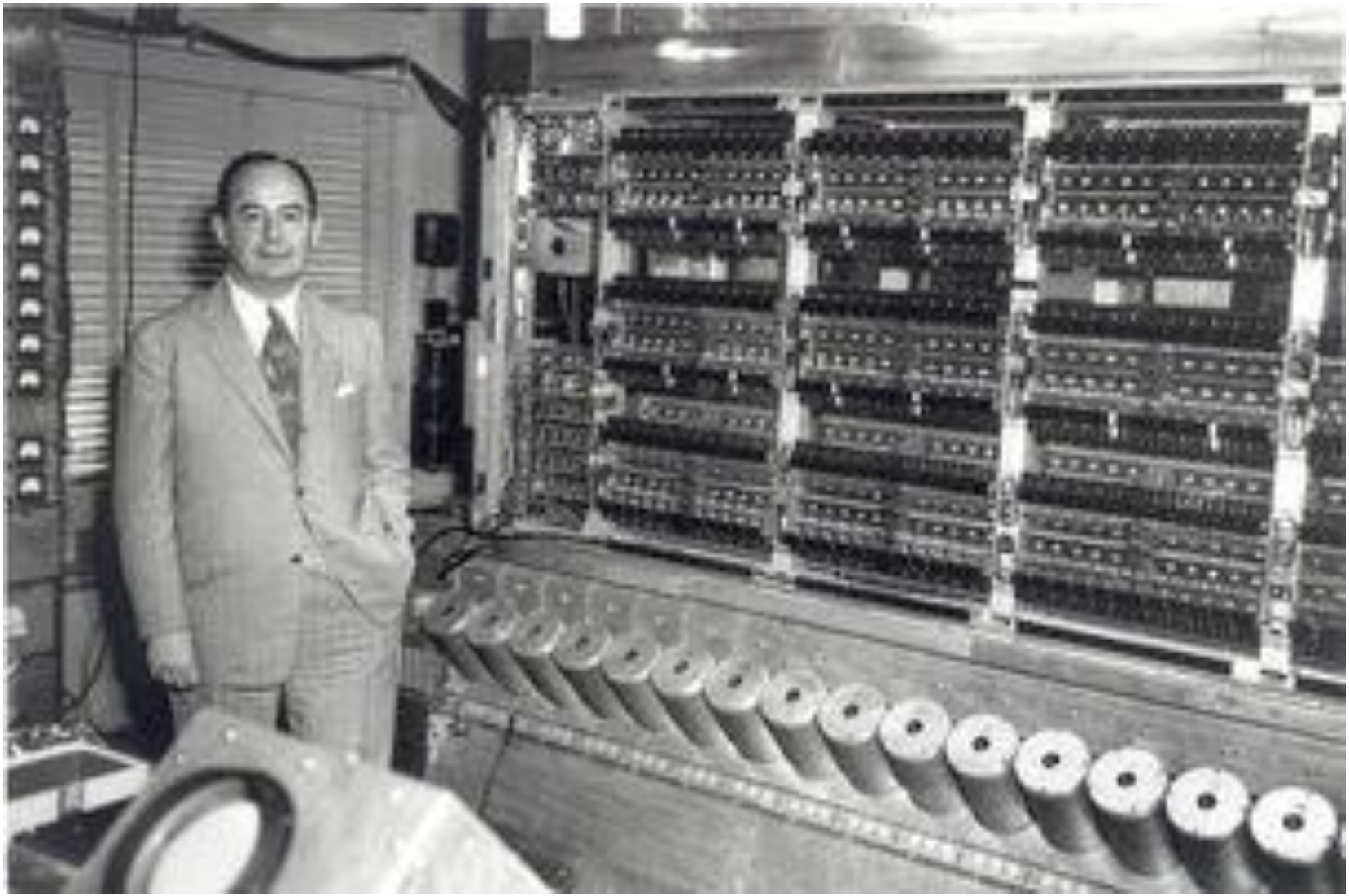



Boarding Carriage
Tail Axis
Anticipating Carriage
Picking-up Sectors
Long Pinions
Bead Axis
Barrel
Reducing Apparatus
Spiral Axis
Central Wheels
Table Axes
Egress Axis (Multiplier/Quotient)
Ingress Axis
Counting Apparatus
Store Axes

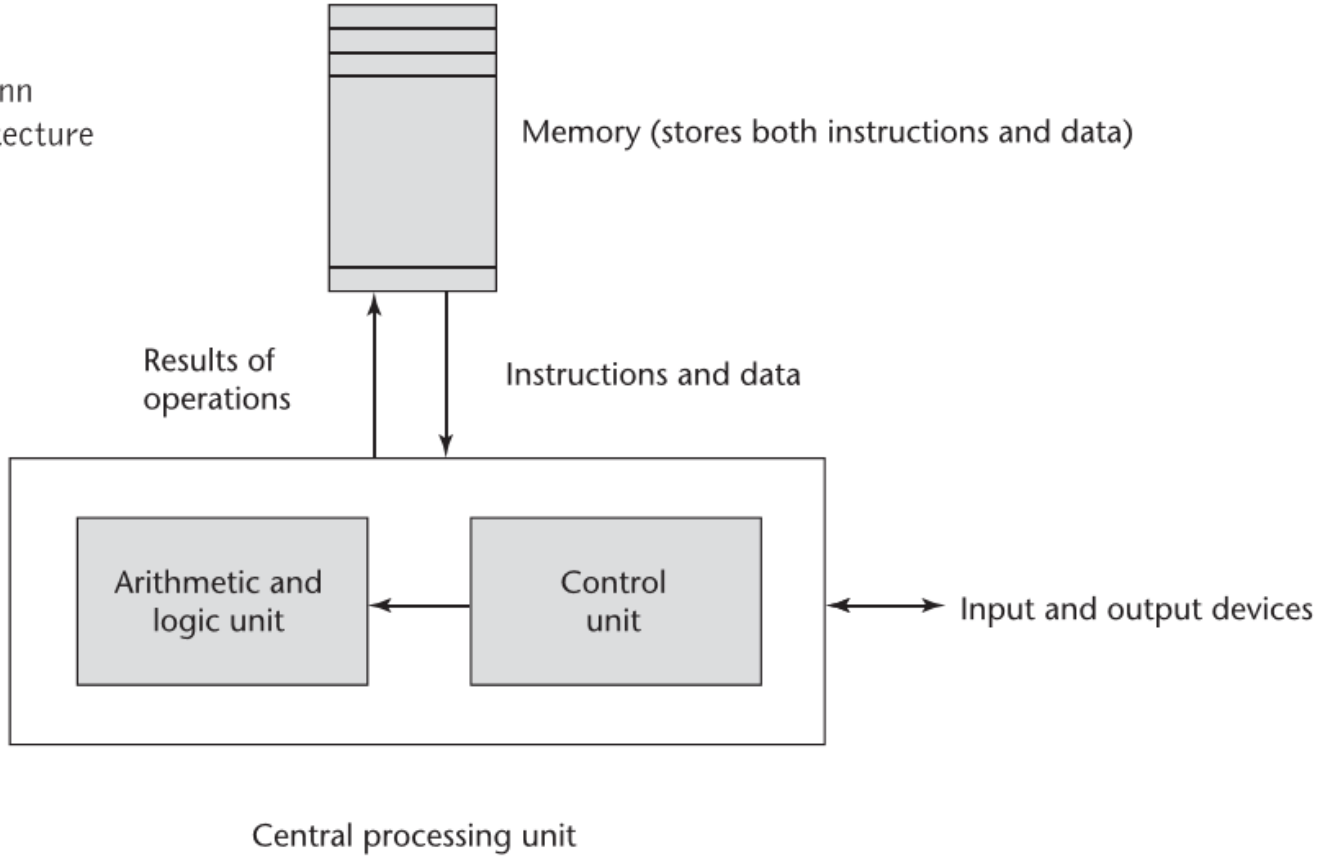
ENIAC – Electronic Numerical Integrator and Computer



John Von Neumann



The von Neumann computer architecture



Important of Programming Languages

Programming language is the heart of software. Without programming we cannot make many applications and software. Programming Language is a key factor of software as well as embedded systems. Without programming language we cannot communicate with machines or systems. Systems only know machine code. Machine codes mean some set of series of numbers. Machine code we can call bits.

Humans only know high level languages but machines do not know high level languages.
Humans and machine could not communicate directly. We need one intermediate because humans could not understand machine languages like machines could not understand high level languages.

Scientific Applications

- The first language for scientific applications was Fortran
- ALGOL-60 and most of its descendants

Business Applications

- The first high-level language for business was COBOL
- Initial versions appeared in 1960.

Artificial Intelligence

- Characteristics were symbolic rather than numeric computations
- The first widely used AI applications was the functional language LISP.
- Logic programming also appeared, the Prolog language.

Systems Programming

- The UNIX operating system is written almost entirely in C

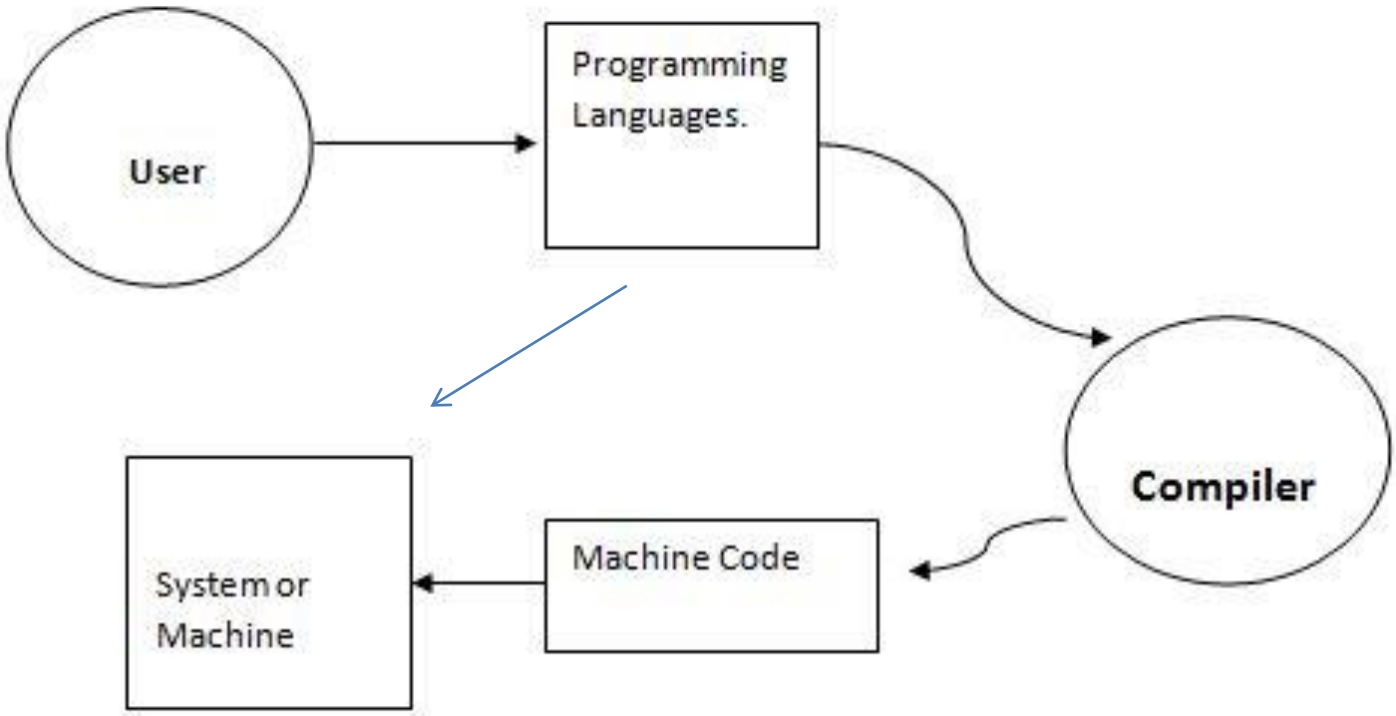
Web Software

- HTML, which is not a programming language, to general-purpose programming languages, such as Java.
- Such code is often in the form of a scripting language, such as JavaScript or PHP.

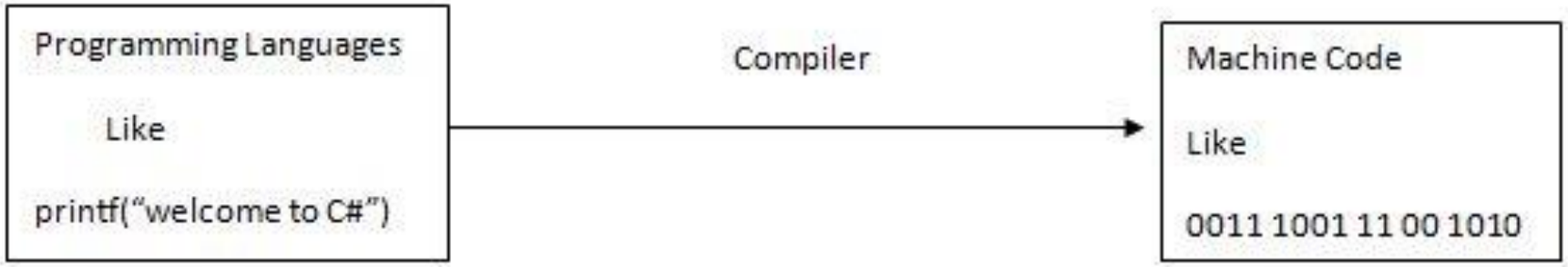
Compiler

Compiler is an intermediary to understand each language. Compiler converts High level languages to low level languages as well as low level languages convert to high level languages. Each language needs at least one compiler. Without compiler we could not understand low high level language.

Flow of Programming Language

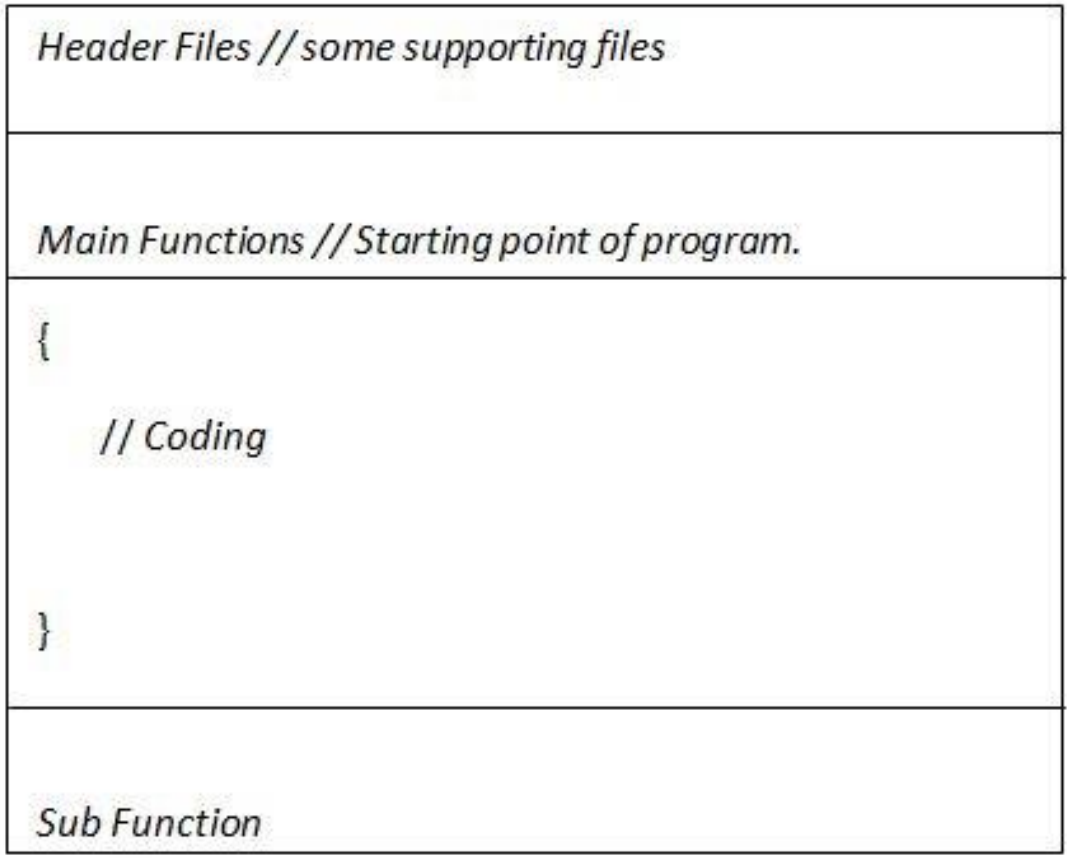


Compiler Task



The user and system code together are sometimes called a load module, or executable image. The process of collecting system programs and linking them to user programs is called linking and loading, or sometimes just linking. It is accomplished by a systems program called a linker.

Structure of Programming Language



Header file is some supporting files. It is located at the top of program. Header file is the head of program. We call header file a different name in different languages. Like bellow.

Header File -> C Language

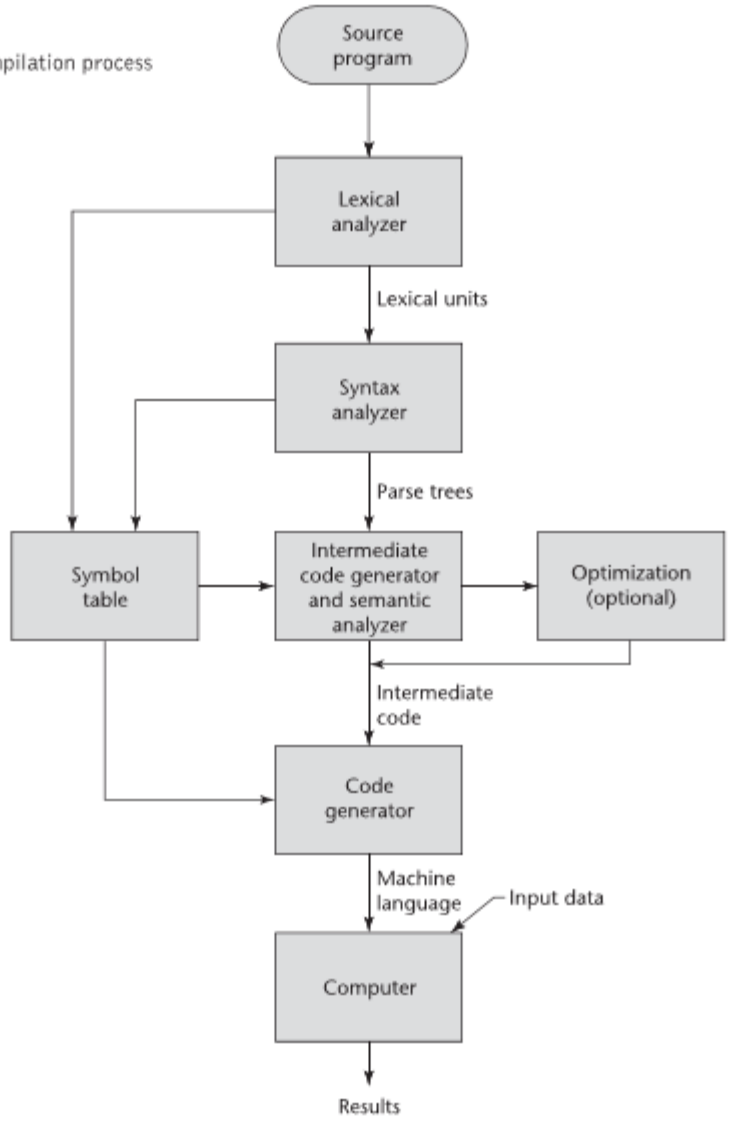
Header File -> C++ Language

Package -> Java Language

Namespace -> C# Language

Main Function is important part of programming languages. Main function is like our body, each and every function happens in main function section. Main function is starting point of programming languages. Sub function is optional one. If need it we can use, otherwise leave it.

The compilation process



von Neumann bottleneck

The speed of the connection between a computer's memory and its processor usually determines the speed of the computer, because instructions often can be executed faster than they can be moved to the processor for execution.

This connection is called the von Neumann bottleneck; it is the primary limiting factor in the speed of von Neumann architecture computers.

All programming language is syntax wise different apart from others, these are same. For example ifI need to print one line using any program using below.

printf(“Welcome To C# Corner”)	C Language
cout<<”Welcome To C# Corner”;	C++ Language
System.out.print(“Welcome To C# Corner”);	Java Language
Console.WritLine(“Welcome To C# Corner”);	C# language

Types of Programming Languages

There are different types of programming languages available. We can see below.

- C
- C++
- Java
- C#
- Python
- Ruby

These are mainly using programming languages in current trends. We can use whichever language we feel is better. C, C++, Java and C# are having different syntax only but concept wise all are same. If we know any one language we can learn all languages easily.

The only exception I can think of is Prolog.

Activity 1

In your own words, define a programming language.