Programming Languages Foundations

Computer Science III

Author: Eng. Carlos Andrés Sierra, M.Sc.
cavirguezs@udistrital.edu.co

Lecturer
Computer Engineer
School of Engineering
Universidad Distrital Francisco José de Caldas

2024-III

Outline

UNIVERSIDAD DISTRITAL
FRANCISCO JOSÉ DE CALDAS

Outline

UNIVERSIDAD DISTRITAL
FRANCISCO JOSÉ DE CALDAS

Babbage Machine

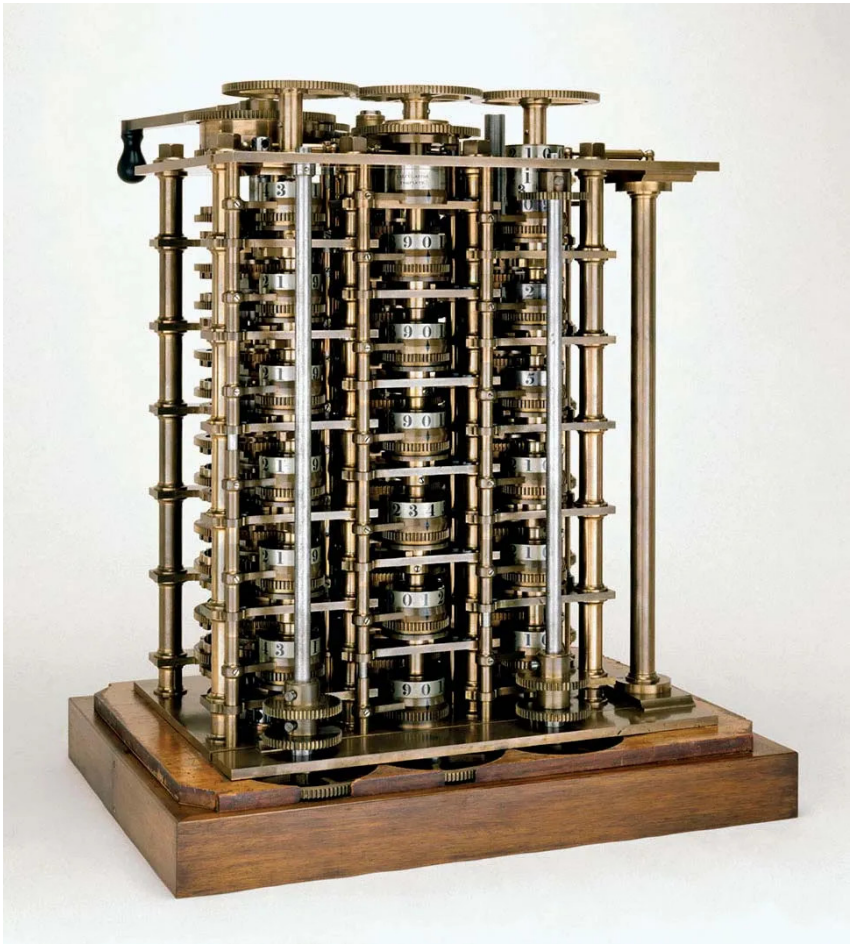


Figure: Analytical Machine

- **Charles Babbage** (1791 — 1871) was an English mathematician, philosopher, inventor, and mechanical engineer.
- He originated the **concept** of a digital programmable computer.
- Considered the "father of the computer". He creates the Analytical Engine.
- The **Analytical Engine** was a general-purpose mechanical computer.

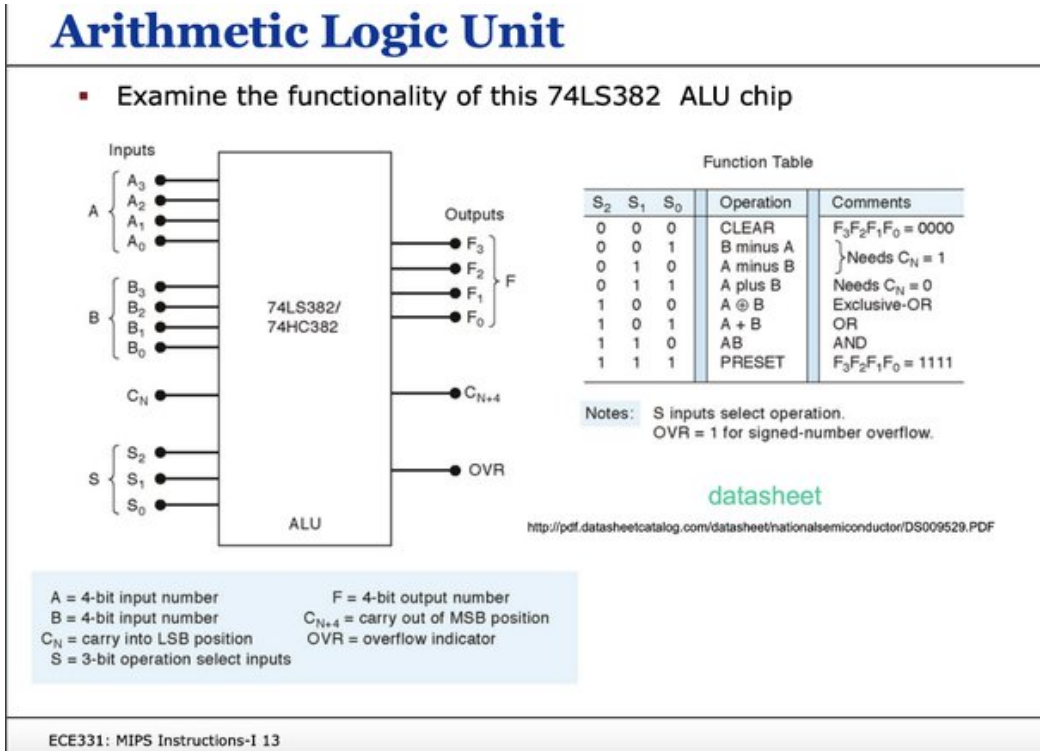
Ada Lovelace

- **Ada Lovelace** (1815 — 1852) was an English mathematician and writer.
- She is known for her work on Charles Babbage's early mechanical general-purpose computer, the Analytical Engine.
- She was the **first** to recognize that the machine had applications beyond pure calculation, and to have published the first algorithm intended to be carried out by such a machine.

Figure: Ada Lovelace

Physical Binary Language

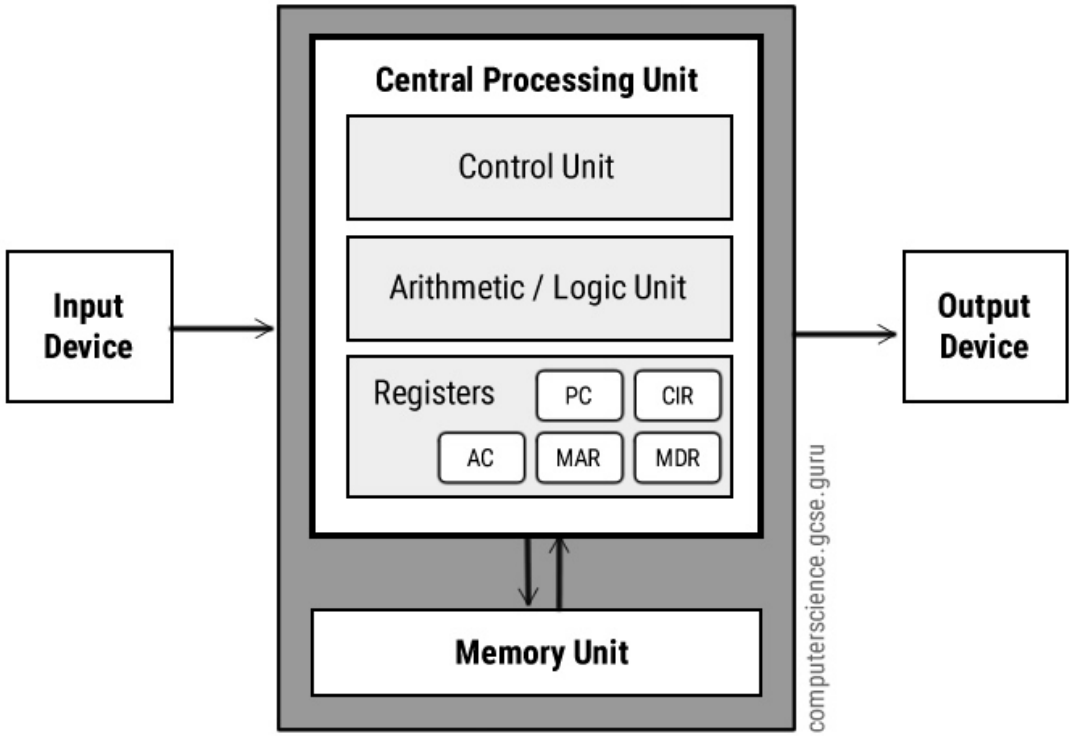
- **Binary** is a base-2 number system. It uses only two symbols: typically 0 (zero) and 1 (one).
- The **bit** is the basic unit of information in computing and digital communications.



Bits and Bytes

Von Neumann Architecture

- The **Von Neumann Architecture** is a computer architecture based on the stored-program computer concept.
- The **design** is based on the concept of an instruction set.
- The **program** and **data** are both stored in the same memory unit.



Memory and Bit Storage

Machine Programming Language

Bit Operations

- **Bitwise operations** are operations that directly manipulate bits.
- They are used in low-level programming for performing calculations, file processing, and data compression.

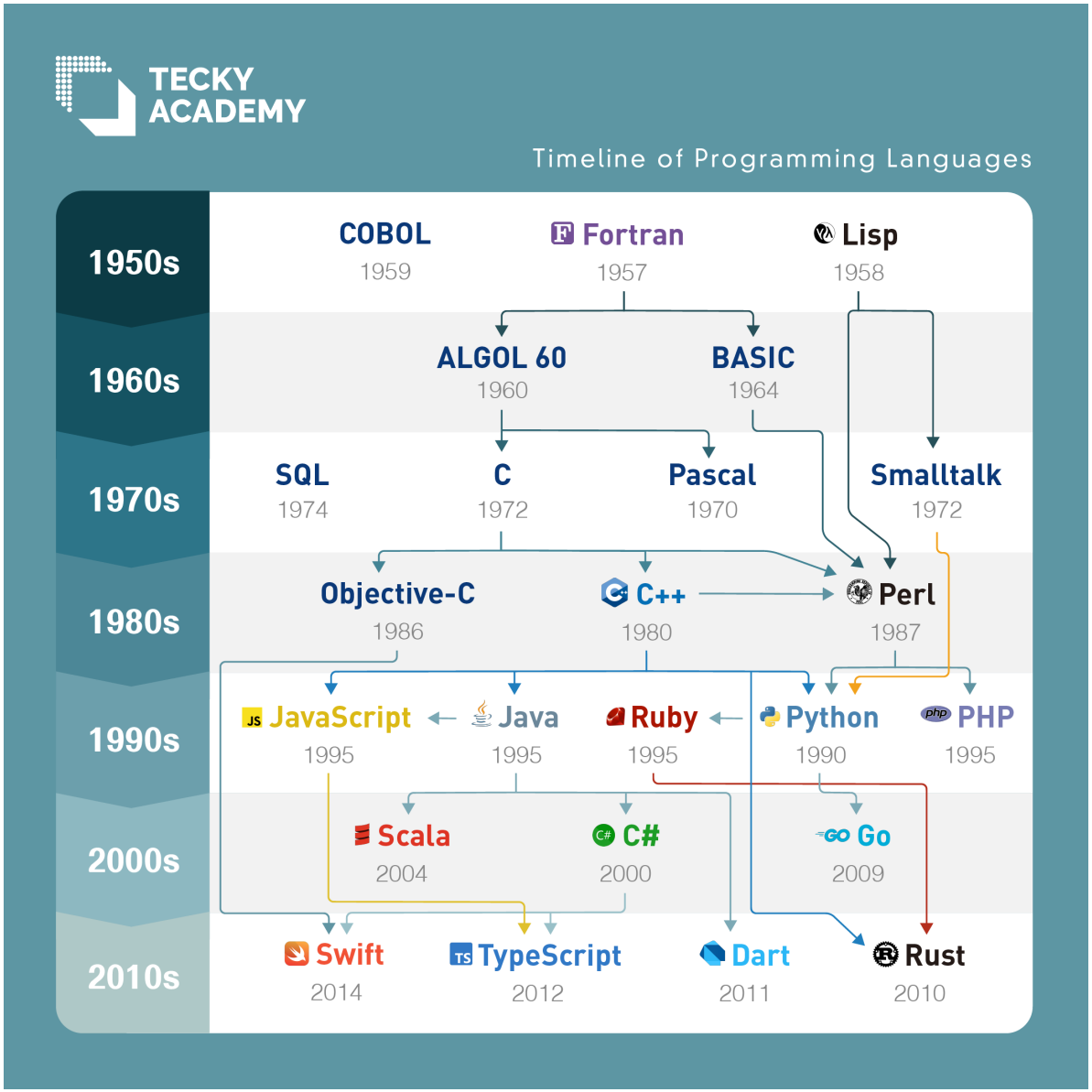
Assembly Programming Language

```
1  ; Example of a basic conditional structure in x86 assembly
2
3  cmp eax, 10     ; Compare the value in eax with 10
4  je equal_label  ; Jump to equal_label if eax is equal to 10
5
6  ; Code for not equal case
7  jmp end_label   ; Jump to end_label to avoid executing the equal case code
8
9  equal_label:
10 ; Code for equal case
11
12 end_label:
13 ; Continue execution
14
```

Listing 1: Basic Conditional Structure in Assembly

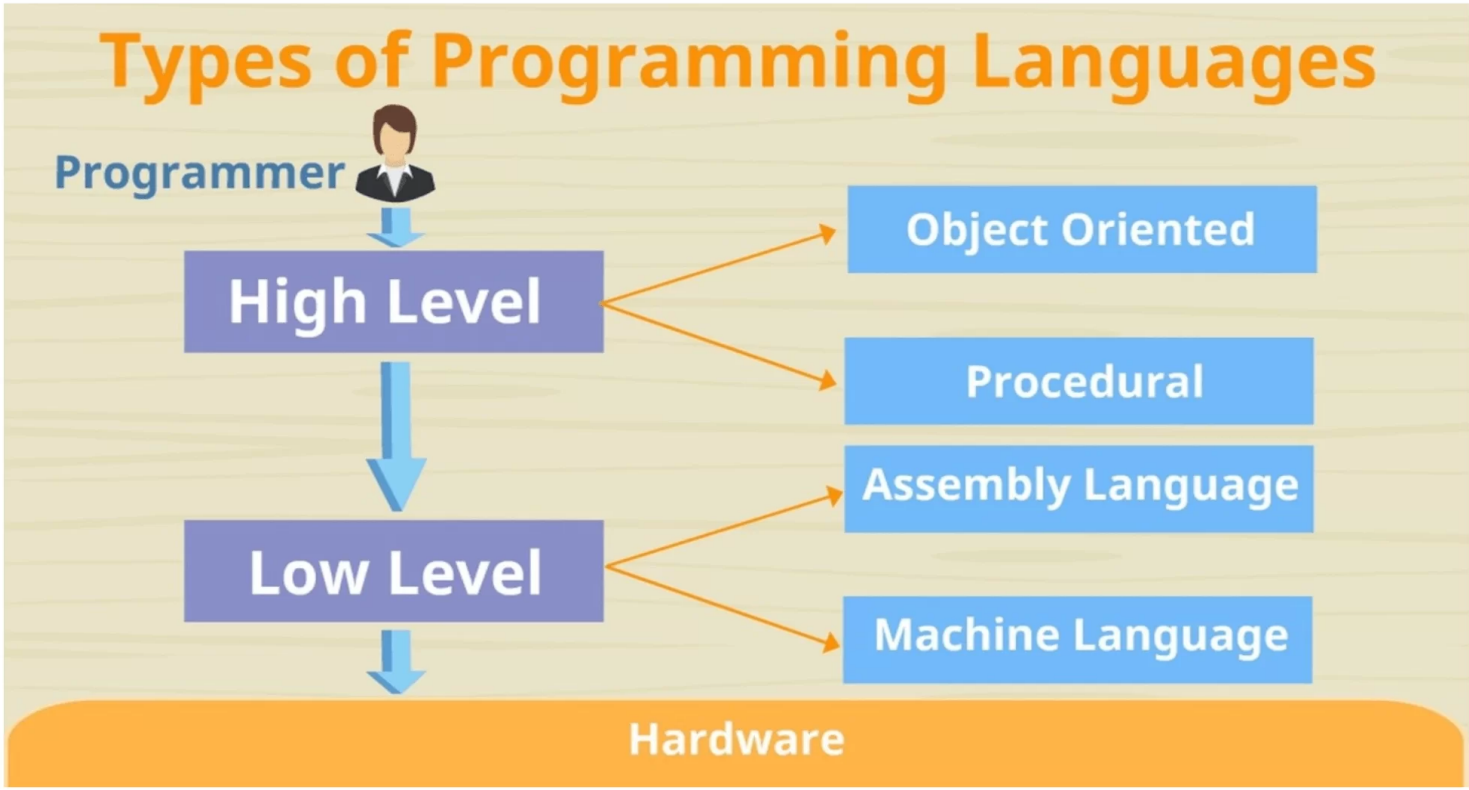
UNIVERSIDAD DISTRITAL
FRANCISCO JOSÉ DE CALDAS

History of Programming Languges



High-Level Programming Languages I

High-Level vs Low-Level



inprogrammer

High-Level Programming Languages II

Purpose of High-Level Languages

Ease of Use

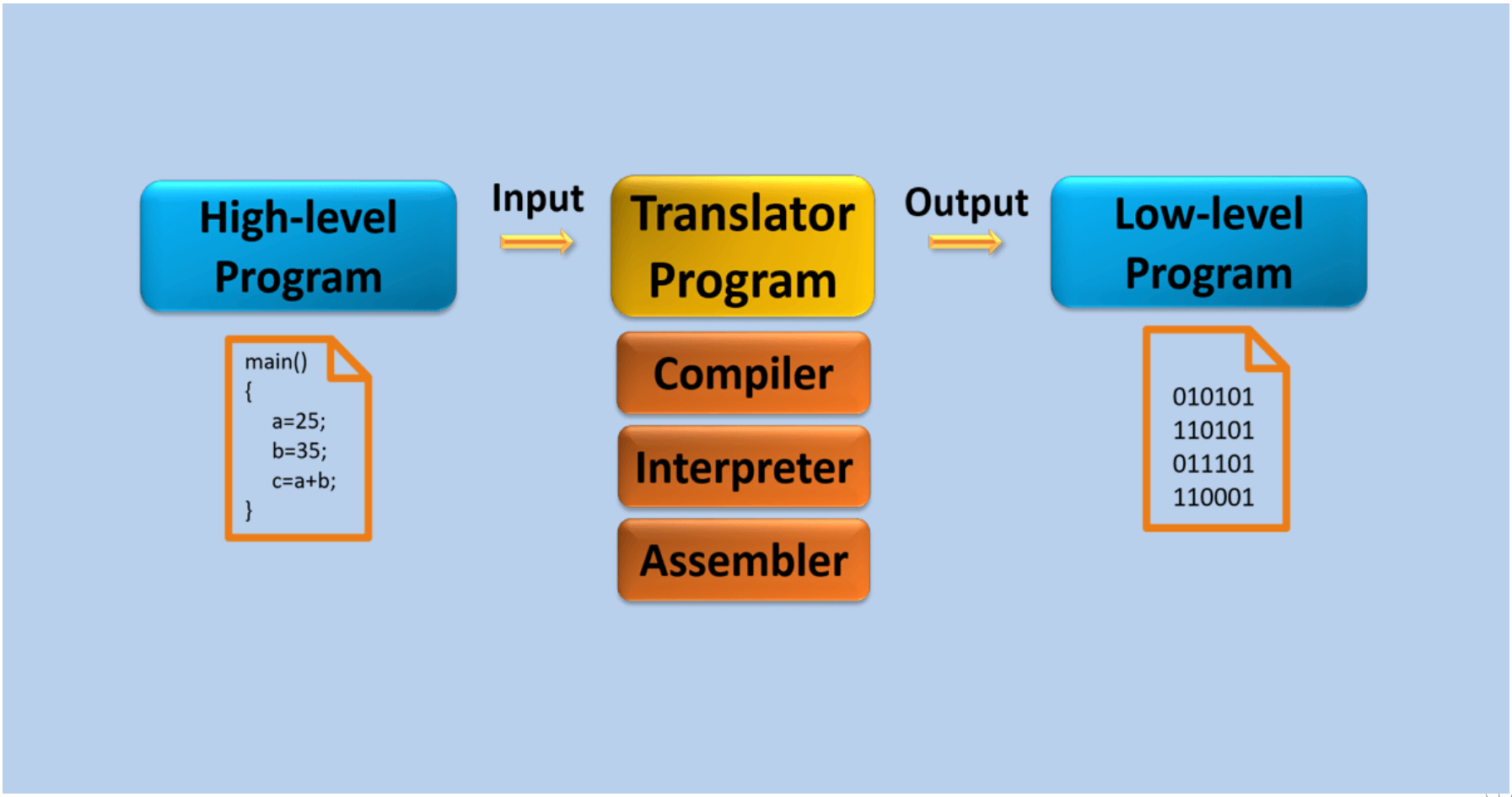
- Simplifying programming
- Minimizing learning curve
- Enhancing productivity
- Automated memory management
- Clear syntax
- Readability and maintainability

Portability Across Systems

- Cross-platform compatibility
- Utilization of compilers and interpreters
- Seamless execution on various platforms
- Reduction of platform-specific modifications
- Enhanced flexibility across environments

Techopedia

Translation Process



Efficiency and Readability

- **Efficiency** is the ability to avoid wasting materials, energy, efforts, money, and time in doing something or in producing a `desired result`.
- **Readability** is the ease with which a human reader can understand the purpose, control flow, and operation of `source code`.

Efficiency and Readability

- **Efficiency** is the ability to avoid wasting materials, energy, efforts, money, and time in doing something or in producing a `desired result`.
- **Readability** is the ease with which a human reader can understand the purpose, control flow, and operation of `source code`.

Outline

UNIVERSIDAD DISTRITAL
FRANCISCO JOSÉ DE CALDAS

Finite State Machines

- A **finite-state machine** (FSM) is a mathematical model of computation.
- It is an abstract machine that can be in exactly one of a finite number of states at any given time.
- The FSM can change from one state to another in response to some external inputs.
- The FSM is defined by a list of states, a list of inputs, a list of transitions, and a list of outputs.

“ ”

.

Turing Master

Finite Automata: Union, Kleene Star

Regular Expressions

Strings Processing

Alan Turing

- **Alan Turing** (1912 — 1954) was an English mathematician, computer scientist, logician, cryptanalyst, philosopher, and theoretical biologist.
- He is widely considered to be the **father of theoretical computer science** and **artificial intelligence**.
- He was highly influential in the development of theoretical computer science, providing a formalization of the concepts of algorithm and computation with the Turing machine.

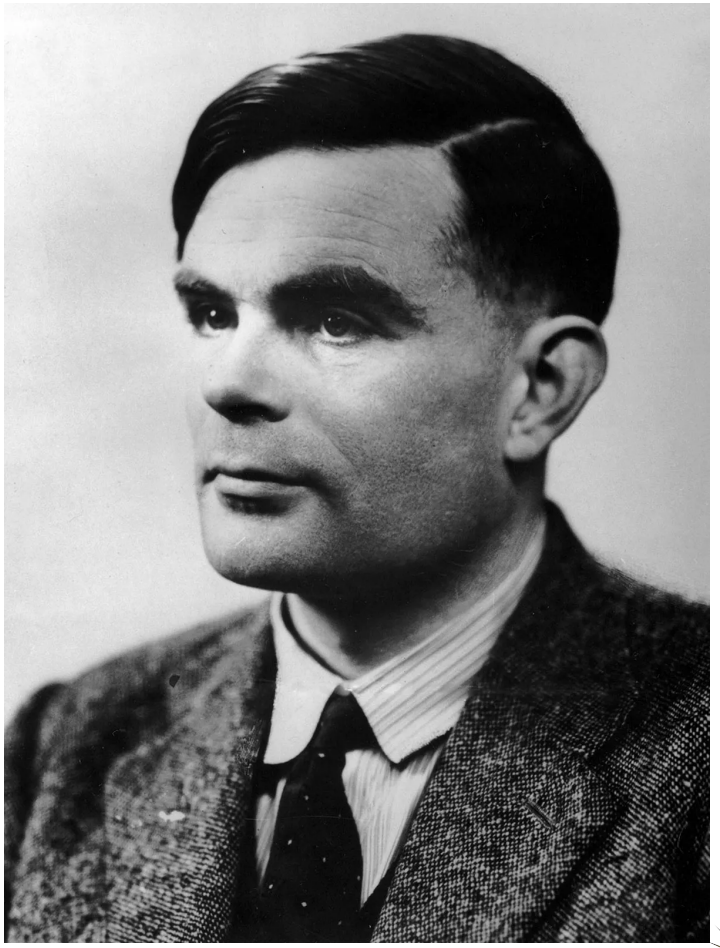


Figure: Alan Turing

Turing Machine

Universal Turing Machine

Outline

UNIVERSIDAD DISTRITAL
FRANCISCO JOSÉ DE CALDAS

Noam Chomsky

- **Noam Chomsky** (1928 —) is an American linguist, philosopher, cognitive scientist, historian, social critic, and political activist.
- He is considered the **father of modern linguistics**.
- He introduced the **Chomsky hierarchy**, a classification of formal languages.

Figure: Noam Chomsky

Natural Processing Language

Formal Languages

Grammars Foundations

Chomsky Formal Norm

Regular Expressions

Equivalence between Gramma and Finite Automatas

Grammars Example

Outline

UNIVERSIDAD DISTRITAL
FRANCISCO JOSÉ DE CALDAS

Thanks!

Questions?

Repo: *https://github.com/EngAndres/ud-public/tree/main/courses/computer-science-iii*

UNIVERSIDAD DISTRITAL
FRANCISCO JOSÉ DE CALDAS