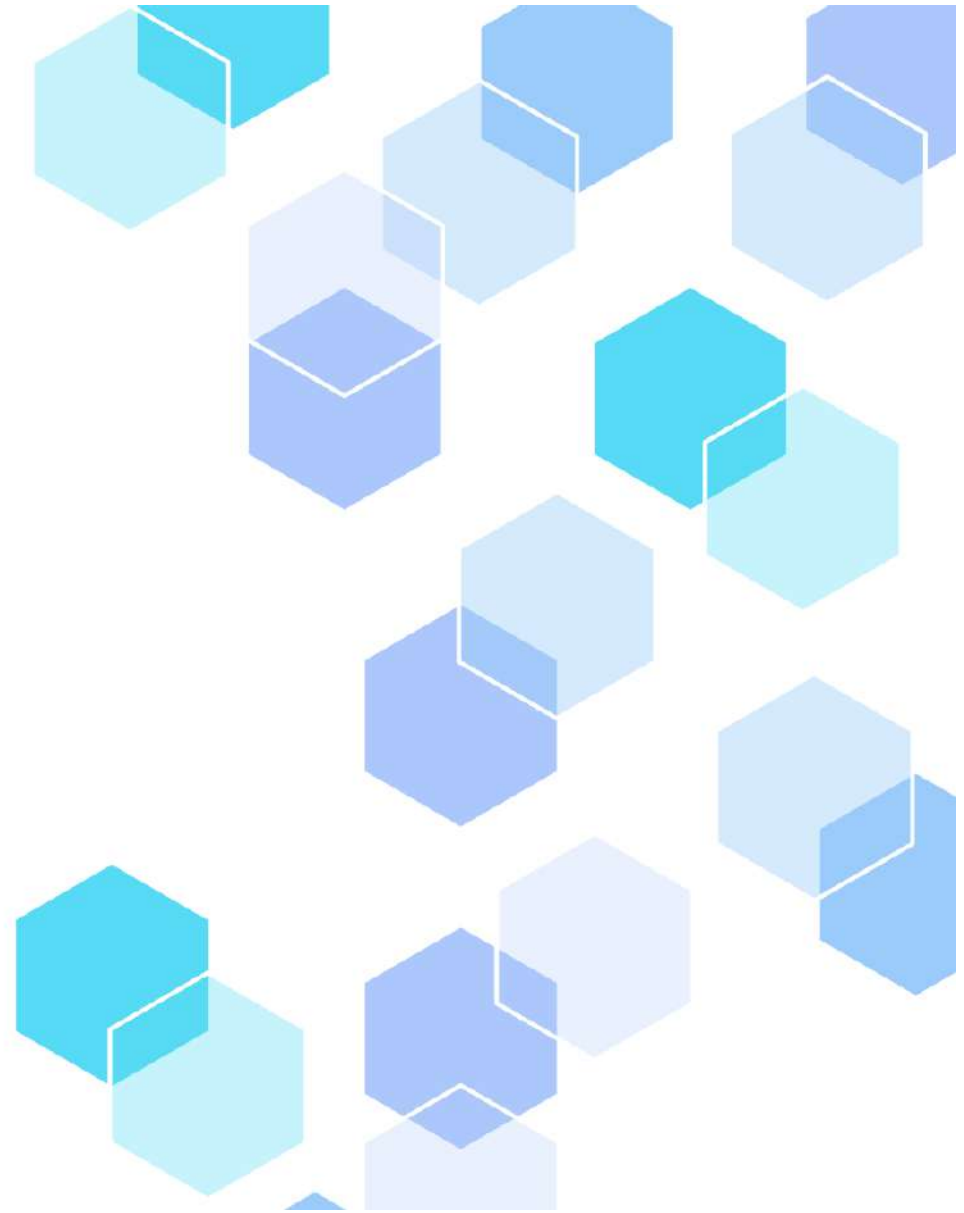


---

# Data Structure and Algorithm

Master the art of DSA  
along with C++

Bring on your coding attitude...





# Table of contents

**01**

## **Programming**

What is Programming?

**02**

## **History of C++**

History & Importance of  
C++

**03**

## **Translator**

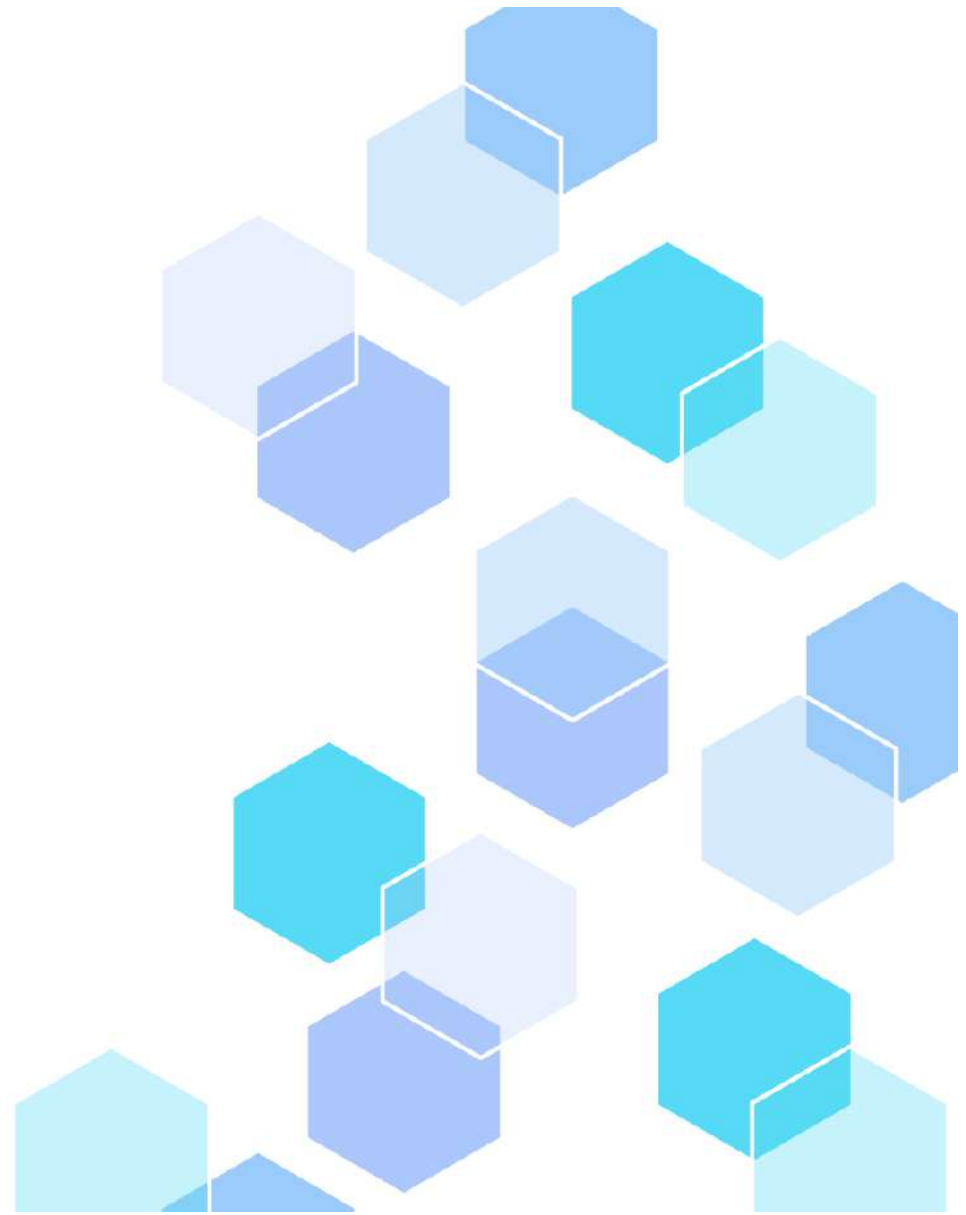
Compiler & Interpreter

---

# 01

# Programming

What is Programming?

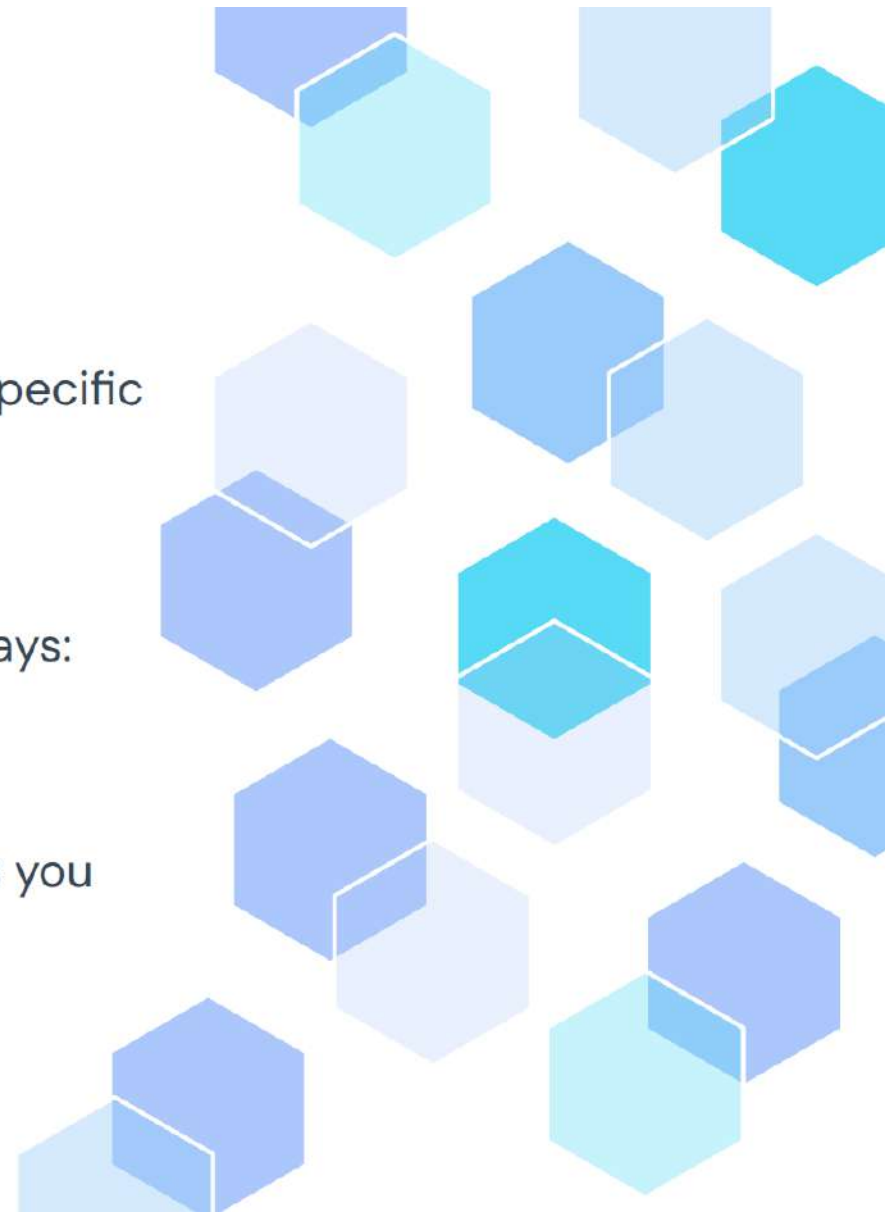


# Programming

A **set of Instructions** for computer to perform specific tasks and solve problems.

Programming impacts your world in countless ways:

- from the websites you visit,
- the apps you use,
- to the games you play and the AI assistants you interact with.



# Steps in Programming



# Choose the weapon

Pick a programming language

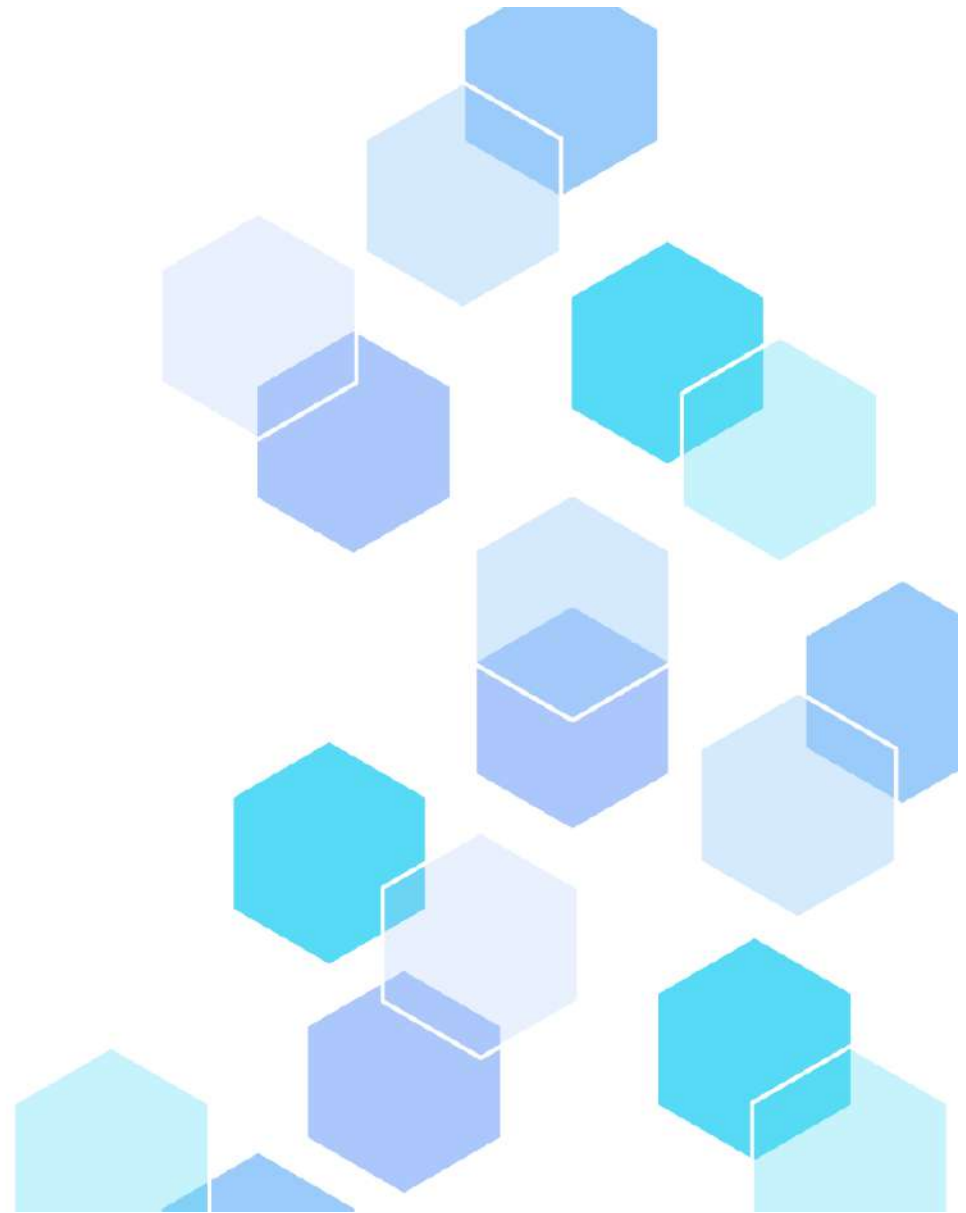


---

# 02

# History

History & Importance of C++





# History of C++

C++ programming language was developed in **1979** by Bjarne Stroustrup at **bell laboratories** of **AT&T** (American Telephone & Telegraph), located in U.S.A.

**Bjarne Stroustrup** is known as the founder of C++ language.



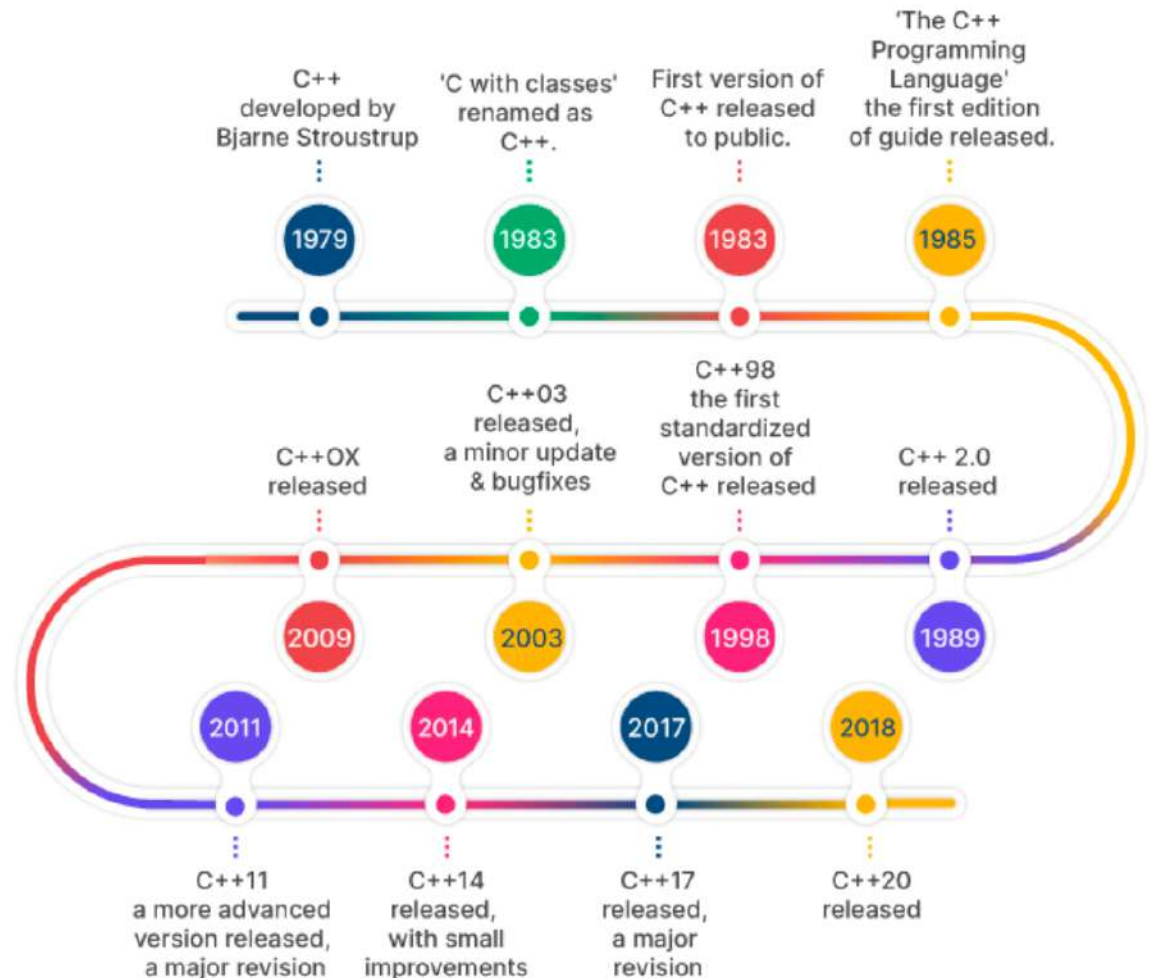


# History of C++

It was developed for adding a feature of **OOP (Object Oriented Programming)** in C language.

C++ programming is "relative" (called a **superset**) of **C**, it means any valid C program is also a valid C++ program.

## History of C++



# Importance of C++

C++ is a **powerful and versatile language** with a crucial role in **various industries**.

Its **performance, versatility**, and established **community** make it a **valuable skill** for programmers seeking to **tackle complex projects** and push the boundaries of software development.



# Importance of C++

**1.**

Performance and  
Efficiency

**2.**

Versatility and  
General-Purpose Nature

**3.**

Object-Oriented  
Programming (OOP)  
Support

**4.**

Large and Active  
Community

**5.**

Legacy Codebase and  
Compatibility

**6.**

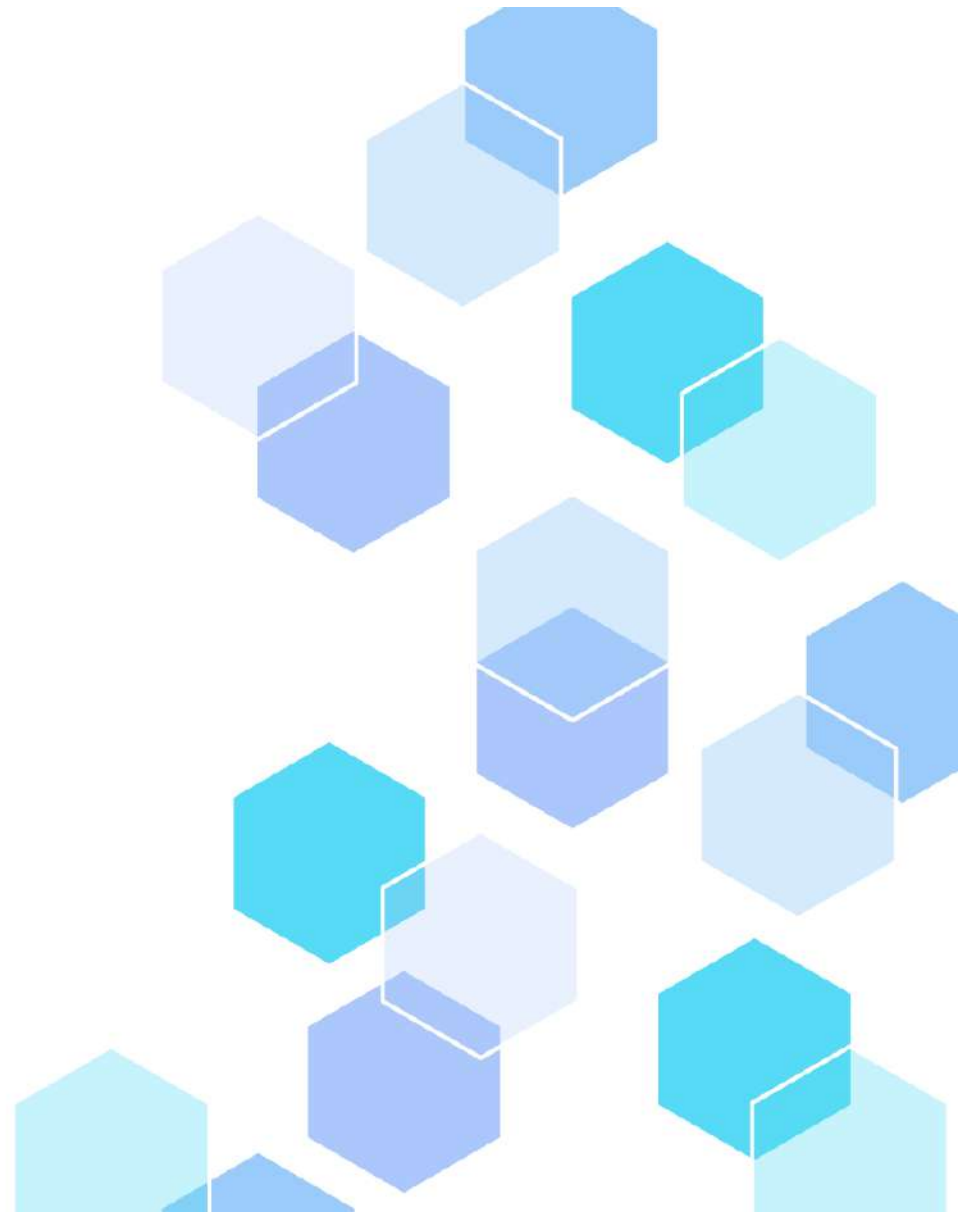
Low-level language  
(System hardware  
accessibility)

---

# 03

# Translator

Compiler & Interpreter



HELLO

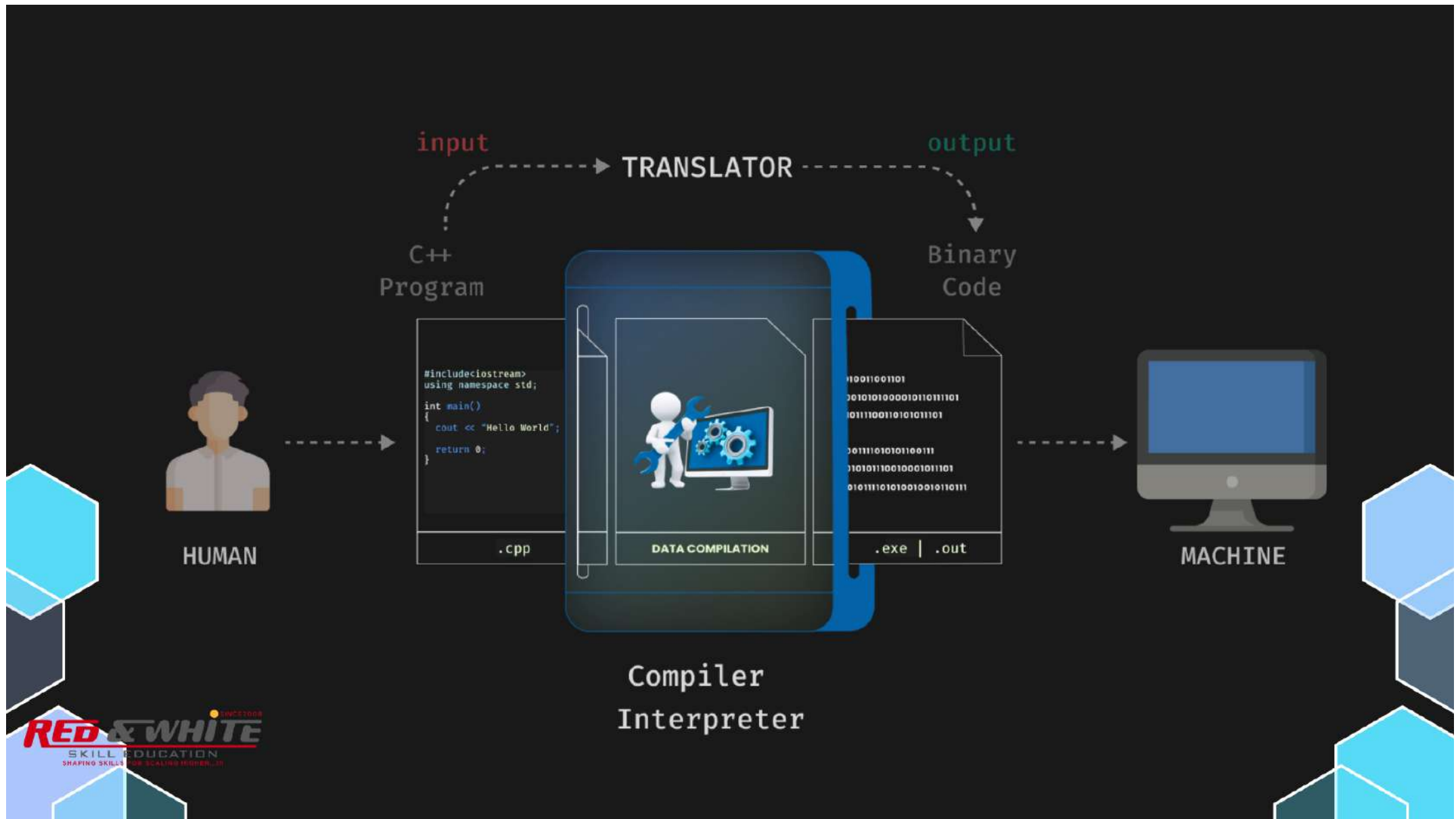
C++

010110

BINARY

TRANSLATOR

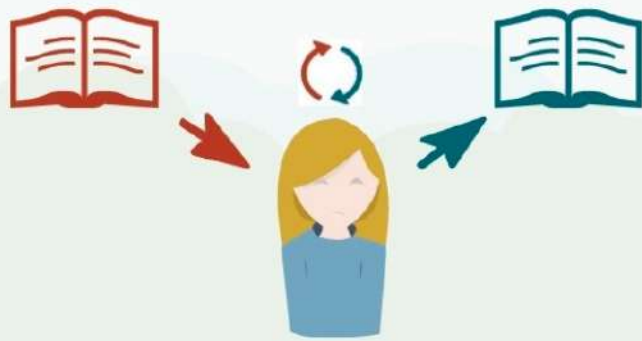






# Types of Translator

## Compiler



Converts **whole program** into binary.

## Interpreter



Converts **line by line** into binary.



C++ is **Compiler based** Programming language.

# Compiler VS Interpreter

COMPARISON	COMPILER	INTERPRETER
Input	It takes an entire program at a time.	It takes a single line of code or instruction at a time.
Output	It generates intermediate object code.	It does not produce any intermediate object code.
Working mechanism	The compilation is done before execution.	Compilation and execution take place simultaneously.
Speed	Comparatively faster	Slower
Memory	Memory requirement is more due to the creation of object code.	It requires less memory as it does not create intermediate object code.
Errors	Display all errors after compilation, all at the same time.	Displays error of each line one by one.
Error detection	Difficult	Easier comparatively
Pertaining Programming languages	C, C++, C#, Scala, typescript uses compiler.	PHP, Perl, Python, Ruby uses an interpreter.

# 10-20-70 Rule for master DSA



**10%**

## Watching

Watching at least one  
lecture daily



**20%**

## Reading

Read 2 hours daily



**70%**

## Practicing

Keep practice 4 hours  
daily

# TL;DR

## History of C++

Founded in 1979

Founded by  
Bjarne Stroustrup

First name was  
"C with Classes"

## Importance of C++

Performance & Efficiency

Versatility and General  
purpose language

Object Oriented  
Programming

# TL;DR

## Compiler

Converts whole program into binary.

Faster than Interpreter

For example, C & C++

## Interpreter

Converts line by line into binary.

Slower than Compiler

For example, Python & Ruby

# Thank you...

Bring on your coding attitude...