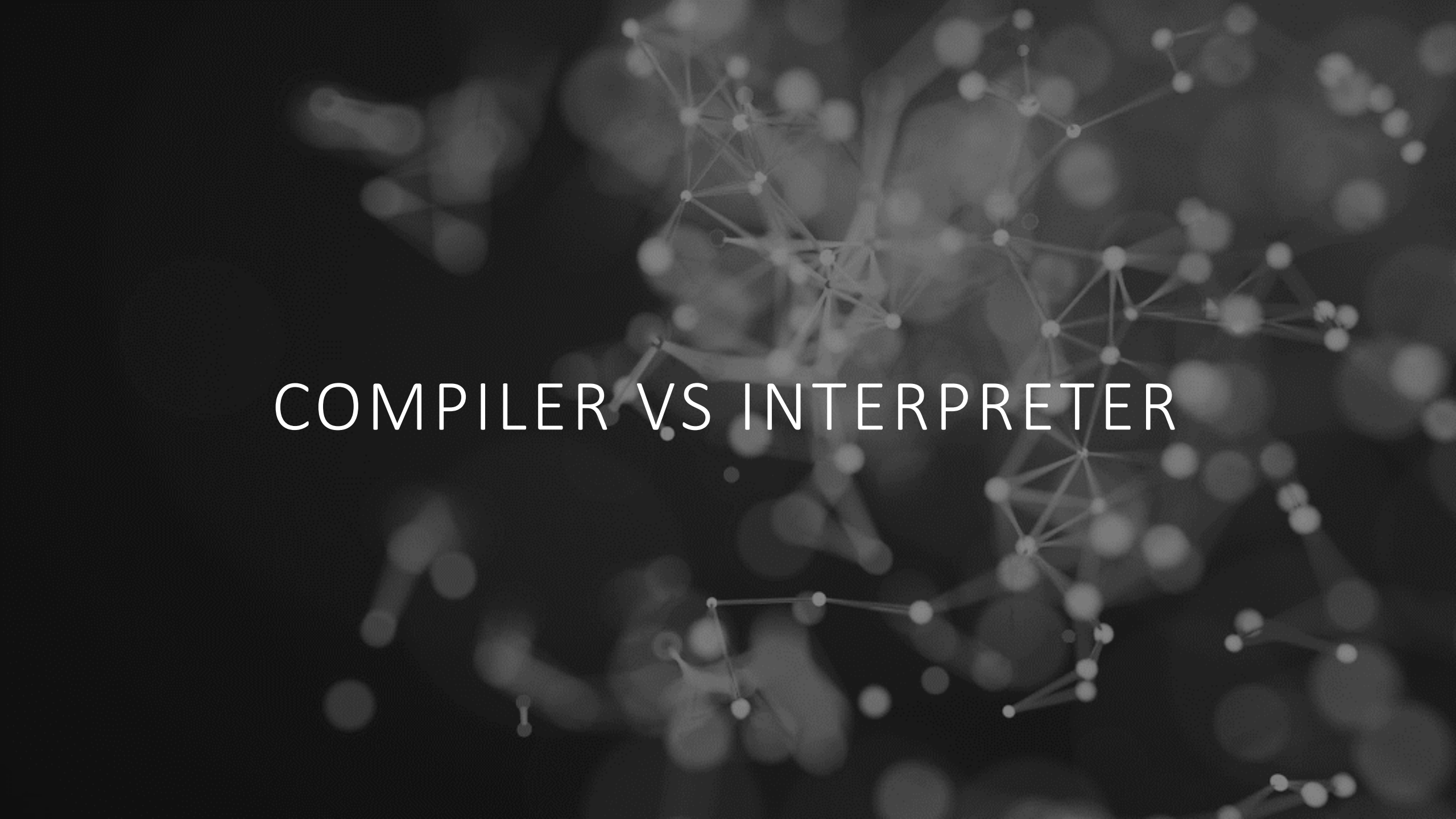
COMPILER VS INTERPRETER

AGENDA

- INTRODUCTION
- WHAT IS A COMPILER?
- WHAT IS AN INTERPRETER?
- KEY DIFFERENCES BETWEEN COMPILER AND INTERPRETER
- ADVANTAGES AND DISADVANTAGES
- CONCLUSION
- REFERENCES & SOURCES

INTRODUCTION

- Programming languages Need Translation To Machine code
- Two Main Translation Methods are Used : Compiler and Interpreter
- This Presentation Compares Them In Terms Of Process, Speed, And Usage

WHAT IS A COMPILER?

The Compiler is a translator that takes input i.e., High-Level Language, and produces an output of low-level language i.e. machine or assembly language. The work of a Compiler is to transform the codes written in the programming language into machine code (format of 0s and 1s) so that computers can understand.

- A compiler is more intelligent than an assembler it checks all kinds of limits, ranges, errors, etc.
- But its program run time is longer and occupies a larger part of memory. It has a slow speed because a compiler goes through the entire program and then translates the entire program into machine codes.

WHAT IS AN INTERPRETER?

An Interpreter is a program that translates a programming language into a comprehensible language. The interpreter converts high-level language to an intermediate language. It contains pre-compiled code, source code, etc.

- It translates only one statement of the program at a time.
- Interpreters, more often than not are smaller than compilers.

KEY DIFFERENCES

COMPILER

- Translates The Entire Program Into Machine Code At Once Before Execution.
- Produces An Executable File (.exe).
- Faster During Execution (After Compilation).
- Errors Are Shown After The Whole Code Is Compiled.
- Used By Languages Like C,C++ And Java

INTERPRETER

- Translate The Program Line By Line During Execution.
- Does Not Produce A Separate Executable File.
- Slower, Because Translation Happens During Runtime.
- Errors Are Shown Immediately, Line By Line.
- Used By Languages Like (Python).

ADVANTAGES

Compiler

- Translates The Whole Code Once , Fast Execution Later.
- Detects Many Errors Before Execution.
- Generates Optimized Machine Code.
- Creates An Independent Executable File.

Interpreter

- Executes Line By Line , Easier For Testing And Debugging.
- No Compilation Time Needed.
- Ideal For Beginners And Scripting Languages.

DISADVANTAGES

Compiler

- Takes More Time To Compile Intially.
- Shows All Errors At Once, Making Debugging Harder.
- Requires Recompilation After Every Code Change.

Interpreter

- Slower Execution Speed.
- Requires Source Code Every Time To Run.
- Not Suitable For Large Programes.

CONCLUSION

In summary, compilers and interpreters both serve the purpose of converting high-level code into something a computer can understand, but they do so in different ways. A compiler translates the whole program at once, which can make it run faster but takes more time to compile. An interpreter translates and runs the code line by line, making it easier to catch errors and debug, though it may run slower.

THANK YOU

Ahmed Gweda Ezzeldin

My GitHub

My Linkedin