Assembly Language

Introduction to Assembly Languages

1 min

An important step in the journey to execute code is Assembly.

The process of code execution starts with a software developer creating a program in a high-level programming language, such as Python or Java, that was designed to be easily read and understood by humans.

For a computer to execute this program, the code must be simplified into

Preview: Docs Loading link description

binary

. In our humble beginnings, programmers initially coded in binary; then, Assembly was created to make it easier for humans to program.

Assembly is a low-level programming language used to directly translate instructions into the computer's

Preview: Docs Loading link description

machine code

in a more human-readable way. Similar to machine code, each instruction begins with an opcode and then references memory locations or

Preview: Docs Data is classified into data types that tell the compiler how the data is intended to be used.

data types

to operate on.

Understanding Assembly is an essential skill on the path to becoming a professional software developer. Studying Assembly allows us to "get under the hood" of our typical programs, making better decisions when it comes to data storage,

Preview: Docs Loading link description

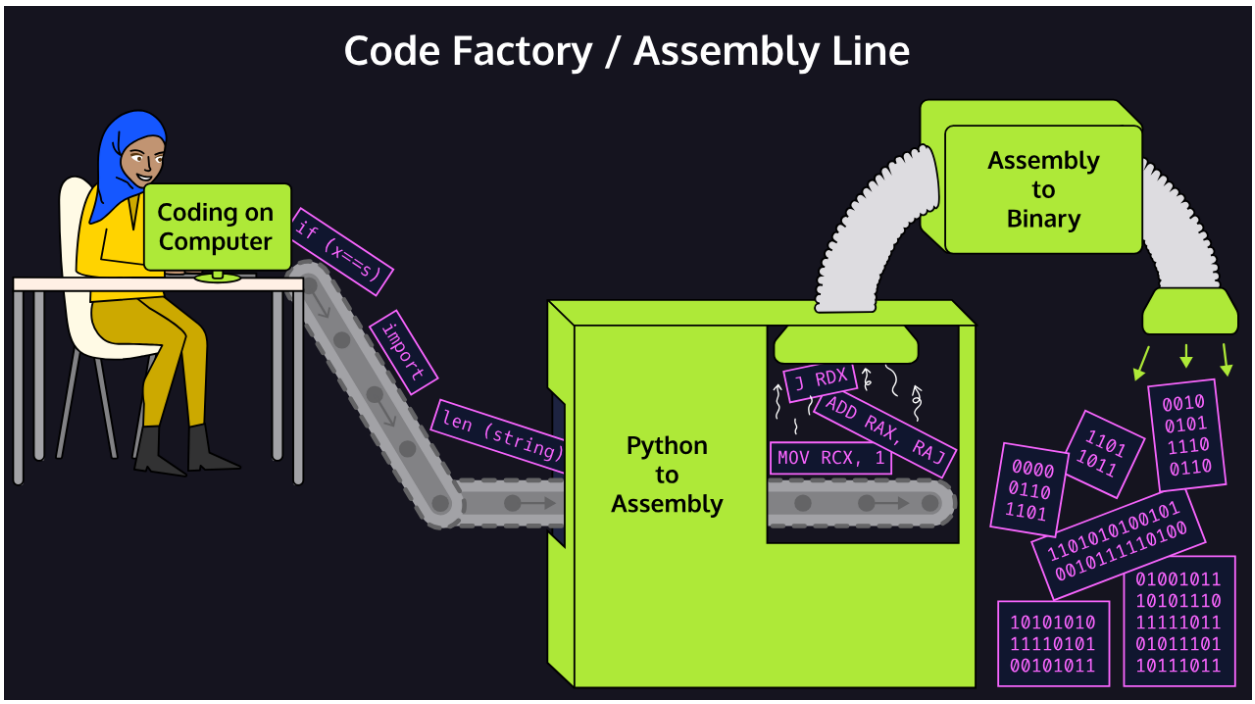
algorithm

optimization, and hardware interactions.

Instructions

Take a look at the Code Factory image to the right. Inside this factory, code is being broken down into simpler and simpler pieces until nothing but binary exists. Each station performs some sort of optimization to create a smoother and more responsive program for the user.

As programs are produced by the software engineer on their IDE, behind the scenes the code is going through a multi-step process in order to make it readable by the hardware.


Code Factory / Assembly Line
Coding on Computer
if (x==5)
import
len (string)
Python to Assembly
J RDX
ADD RAX, RAJ
MOV RCX, 1
Assembly to Binary
0010 0101 1110 0110
1101 1011
0000 0110 1101
1101010100101 0010111110100
01001011 10101110 11111011 01011101 10111011
10101010 11110101 00101011