C++ : Zero to Hero

Introduction to low-level languages

ML and assembly languages are known as Low level language.

Drawback :-

- Low level are architecture specific.
- Hard to understand.

Introduction to high-level Languages

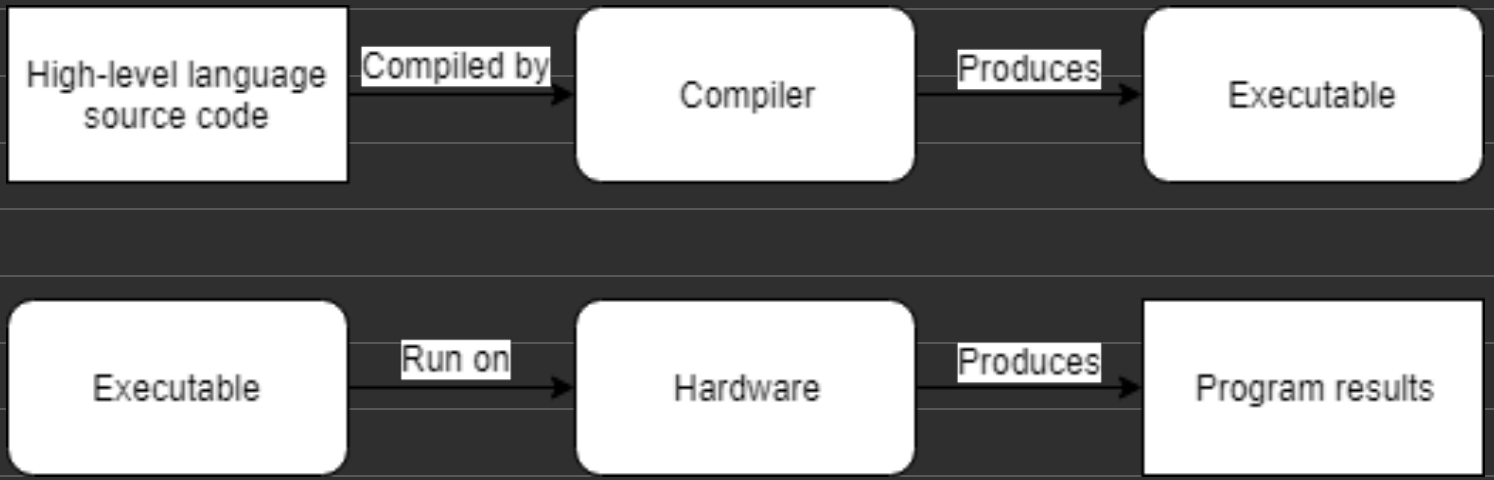
- C, C++, Python, Pascal.

#Compiler :- is a program that reads source code of one language (high level language) and converts it into low level language.
ex) C++ source code into machine code.

Note :- C++ code can be also converted into assembly level code.

Note :- Machine level code that later converted into executable file that can be used by O.S.

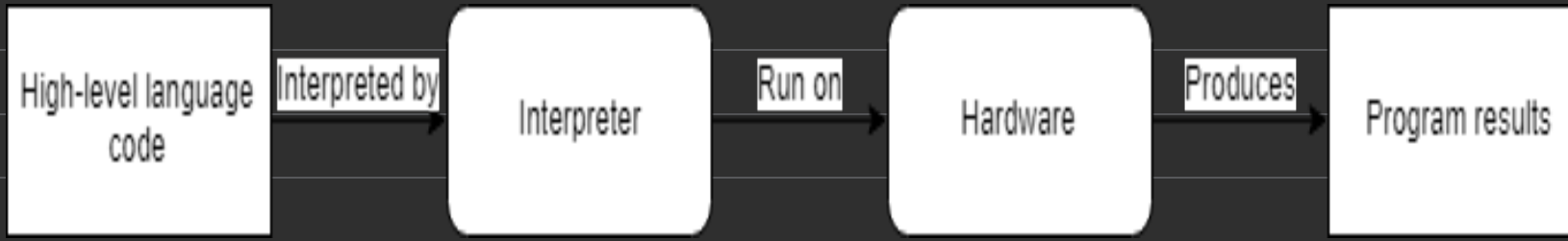
Compiling process



Interpreter

Interpreter is a program that directly executes the instructions in src code without requiring them to compile first.

Process



Comparison :-

Distributing a compiled program doesn't required src code distribution.

C & C++ Philosophy :- Trust the programmer.

C++ Goodness :-

Video games
Real-time systems (e.g. for transportation, manufacturing, etc...)
High-performance financial applications (e.g. high frequency trading)
Graphical applications and simulations
Productivity / office applications
Embedded software
Audio and video processing
Artificial intelligence and neural networks

Error :-

Code > C++ error.cpp > main()

```cpp
#include <iostream>

int main()
{
    std::cout << "Hello world!" return 0;
}
```

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
abhaysoni512@MacBookAir ~/Desktop/cpp/Day1/Code cd "/Users/ab
error.cpp:5:32: error: expected ';' after expression
    5 |     std::cout << "Hello world!" return 0;
      |                                ^
      |                                ;
1 error generated.
✘ abhaysoni512@MacBookAir ~/Desktop/cpp/Day1/Code
```

Random Access Memory

The main memory in a computer is called Random Access Memory (often called RAM for short). When we run a program, the operating system loads the program into RAM. Any data that is hardcoded into the program itself (e.g. text such as "Hello, world!") is loaded at this point.
The operating system also reserves some additional RAM for the program to use while it is running. Common uses for this memory are to store values entered by the user, to store data read in from a file or network, or to store values calculated while the program is running (e.g. the sum of two values) so they can be used again later.

You can think of RAM as a series of numbered boxes that can be used to store data while the program is running.

Object and Variables :-

Object represents a region of storage that can hold a value.

An object with a name is called variable.

Defining a Variable :-

```
int x ;
```

Datatype :- Datatype (Also known as type) determines kind of value object store.