

*Introduction to computer language

*OBJECTIVES:

- TO UNDERSTAND WHAT IS COMPUTER LANGUAGE
- TO UNDERSTAND LOW LEVEL AND HIGH LEVEL LANGUAGE
- TO DIFFERENTIATE BETWEEN LOW LEVEL AND HIGH LEVEL LANGUAGE

* Computer languages

A computer language is an **artificial language** designed to work for different applications in different environments.

Programming languages can be used to create programs that control the behavior of a computer and serve any purpose.

* Computer languages

The term programming language usually refers to high-level languages such as BASIC, C, C++, COBOL, FORTRAN, Pascal, Java, Oracle, ASP.NET, VB.NET and C#.

Each language has a unique set of keywords (words that it understands)



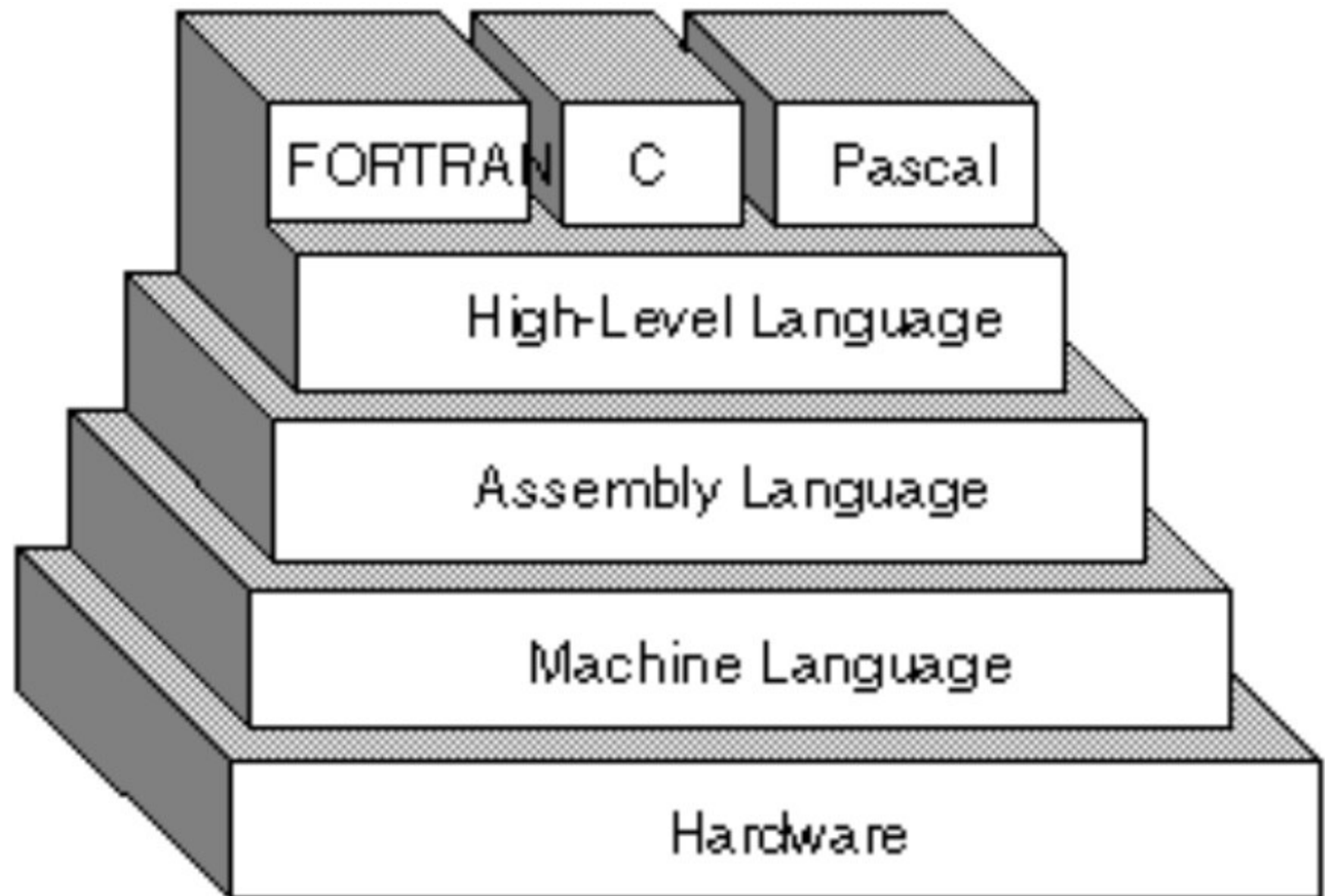
Computer Languages

Human languages are known as natural languages. Unfortunately, computers can not understand natural languages, as a result we must communicate with computers using computer languages.

These languages are:

- **High Level Languages**
- **Low Level Languages**
- **Assembly Language**
- **Machine Language**

* Architecture



* High level languages

It is a set of words and symbol which a programmer uses to write a program

High-level languages are much closer to human language. A programming language such as C, FORTRAN or Pascal that enables to write programs which is understandable to programmer (Human) and can perform any sort of task, such languages are considered high-level because they are closer to human languages.

High level language must use interpreter, compiler or translator to convert human understandable program to computer readable code (machine code).

There are many high level languages

Some Examples:

COBOL	Business applications
FORTRAN	Engineering & Scientific Applications
PASCAL	General use and as a teaching tool
C & C++	General Purpose - currently most popular.
PROLOG	Artificial Intelligence
JAVA	General all purpose programming
.NET	General or web applications.

Advantages of High level language over low level lang:

- They are near to English language, that is they are easier to read, write and maintain.
- High-level languages make complex programming simpler.
- High level languages is portable, i.e., they can work on different operating system.
- Length of the program is also small compared with low level.
- Many real time problems can be easily solved with high level language.

* Disadvantages of High level languages

- They need to be translated for the computer to understand, hence work slower than machine code.



Low Level Language

A computer low level language that deals with hardware registers by name is known as **assembly language**.

Assembly language is the best example of low level language, it is in between machine language and high-level language.

A low-level language does not need a compiler or interpreter to run the program, the processor run low-level code directly.



* Assembly Language

Assembly languages have the same structure and set of commands as machine languages, but they enable a programmer to use names instead of numbers.

In the early days of programming all programs were written in assembly language but now almost programs are written in a high-level language.

Each assembly language is specific to a particular computer architecture, in contrast to most high level programming languages, which are generally portable across multiple systems.

* Is low level language still in use?

- * Programmers still use assembly language when speed is essential or when they need to perform an operation that isn't possible in a high-level language.

* Some examples where low level languages are still used

- Writing games or simulation program
- Writing operating systems
- Programming robots
- Computers viruses, certain device drivers or other items very close to the hardware or low-level operating system.



Assembly Language

It uses **mnemonic codes** (short forms) for instructions and allows the programmer to introduce names for blocks of memory that hold data.

Assembly language is designed to be easily translated into machine language.

Examples:

MOV AL, 1h ; Load AL with immediate value 1

SUB R0,R0,#1 ;Decrement R0

MUL R0,R1,R2

STR R0,[R1,#20]

* Machine language

Machine code or machine language is a system of instructions and data executed directly by a computer's CPU. The lowest-level programming language that only be understood by computers.

Computer language that is directly executable by a computer without the need for translation by a compiler or an assembler.



Machine Language

The native language of the computer,

The set of symbolic instructions in binary that is used to represent operations and data in a machine called machine code

Machine Language: “0110101100101000”

machine language is a collection of binary digits or bits that the computer reads and interprets.

Machine language is the only language a computer understands. It is almost impossible for humans to use because they consist entirely of numbers.



Program code converters

There are four models of execution of program by processor in high-level languages:

- **INTERPRETER**
- **COMPILER**
- **ASSEMBLER**
- **TRANSLATOR**



Interpreter

An interpreter is a program that executes programming code directly.

Interpreter can convert a source code , usually on a step-by-step, line-by-line and unit-by-unit basis into machine code.

Interpreted languages are read and then executed directly, with no compilation stage.



Compiler

Compiler is a program that compile source code into executable instructions that a computer can understand, it compiles code and produce in **EXE** format before it run.



Assembler

Assembler normally converts assembly language's source code into machine language.

Assembly language is a family of low-level languages for programming computers. It implements a symbolic representation of code.

An assembler is a program that takes basic computer instructions and converts them into a pattern of **bits (0 & 1)** that the computer runs to produce result.



Translator

Translator is a computer program that translates one programming language instruction(s) into another programming language instruction(s) ...

A computer source code is to be translated into a low-level programming language where native code compilers change it in machine language for execution.

The source code is the code written either in assembly or high level language.

* Questions

- How many categories of computer languages are there and name them?
- Explain briefly the differences between each categories.
- State the different types of translators and there functions.
- State 2 programming languages in high level.
- Is low level languages still used?
- State one application where low level language is still used.

* Questions from past exams

- June 2006 Question 6

- (a) Give 2 benefits of using a high level language for writing programs
- (b) State one type of program that would be written in a low level language rather than a high level language and give a reason why.

- Nov 2007 Question 2

State two differences between high level languages and low level languages

To be submitted on:

* END OF PRESENTATION

***THANKS** YOU FOR YOUR ATTENTION