

Lecture 03

Program

Mostafa Elhosseini

Professor at College of Computer Science and
Engineering

Taibah University

<https://youtube.com/drmelhosseini>

| What is Programming?

- A **program** is a sequence of instructions that specifies how to perform a computation on computer hardware
 - Input
 - Output
 - Math
 - Decision
 - **Loop**
Perform an action repeatedly, usually with some variation

| What is Programming?

- A program makes a computer usable.
- Without a program, a computer, even the most powerful one, is nothing more than an object.
 - Without a player, a piano is nothing more than a wooden box
- Computers can only evaluate the results of very **fundamental operations**, like **adding or dividing**, but they can do it **very fast**, and can **repeat** these actions virtually any number of times

| How does a Computer work?

- Example: **Average speed**
 - Accept a number representing the distance;
 - Accept a number representing the travel time;
 - Divide the former value by the latter and store the result in the memory;
 - Display the result (representing the average speed) in a readable format.
- These four simple actions form a program
 - Of course, these examples are not formalized, and
 - They are very far from what the computer can understand, but...
 - They are good enough to be translated into a language the computer can accept. >>> Keyword

| How does a Computer work?

- Computers have their **own language**, too, called **machine language**, which is very rudimentary
 - Well-trained **dog**
 - It responds only to a **predetermined set of known commands**
 - Take that number, divide by another and save the result
 - A complete set of known commands is called an **instruction list**, sometimes abbreviated to **IL**

| Machine Language

- Any **computer** can directly **understand** only **its own machine language**, defined by its **hardware design**.
- **Machine languages** generally consist of strings of numbers (ultimately reduced to **1s and 0s**) that instruct computers to perform their most elementary operations one at a time.
- Machine languages are machine dependent

| Assembly Language

- Instead of using the strings of numbers that computers could directly understand, programmers began using **English-like abbreviations** to represent elementary operations
- **Translator** programs called **assemblers** were developed to convert early **assembly-language** programs to **machine language** at computer speeds.

| High-Level Language

- We need a language in which **humans can write** their programs and a language that **computers may use to execute the programs**
 - one that is far more complex than machine language and yet far simpler than natural language.
 - Such languages are often called **high-level programming languages**.
- A program written in a **high-level programming** language is called a **source code** (in contrast to the machine code executed by computers). Similarly, the **file** containing the source code is called the **source file**.

| High-Level Language

- To speed the programming process, high-level languages were developed in which single statements could be written to accomplish substantial tasks
- Translator programs called compilers convert high-level language programs into machine language.
- High-level languages allow you to write instructions that look almost like everyday English and contain commonly used mathematical notations

| References

- Java: How to Program, 9th Edition (Deitel) 9th Edition by Paul Deitel , Harvey M. Deitel
- Head First Java, 2nd Edition 2nd Edition by Kathy Sierra, Bert Bates
- Think Java: How to Think Like a Computer Scientist 1st Edition by Allen B. Downey, Chris Mayfield