* Computer Architecture:

Computer architecture refers to the design and organization of a computer's components, including its hardware and the way these components interact to perform various tasks. It involves the structure and behavior of a system, focusing on how the CPU, memory, input/output devices, and other hardware components are interconnected.

**

Why we should learn computer architecture?

Ans:

Learning computer architecture is valuable for many reasons:

1. Understanding How Computers Function: It helps us to understand how a computer operates, rather than just using it. Also, we will learn how it processes data, runs programs, and connects with other devices.

2. Improving Performance: By understanding how computer work, we can create more efficient software that takes full advantage of the hardware, making it run faster and smoother.

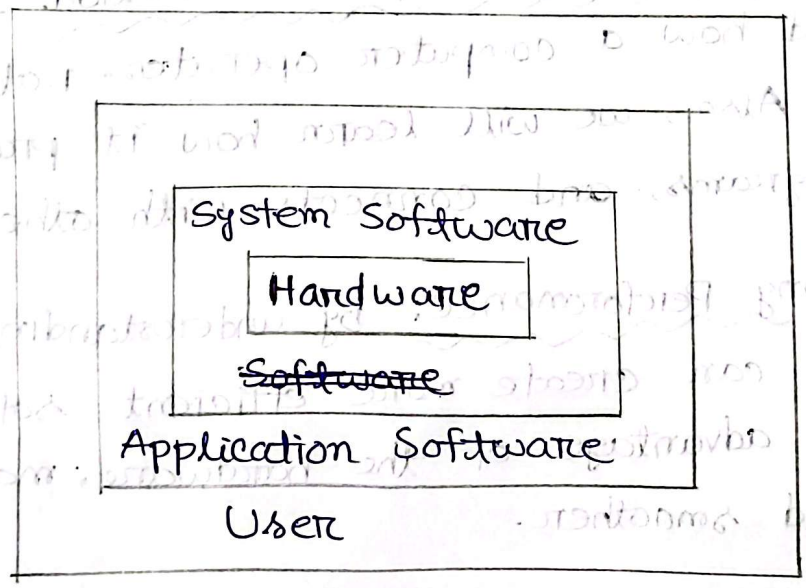
~~3. Fixing Issues: knowing~~

3. Designing Better Hardware: If we want to create or improve computer systems, understanding architecture helps us build hardware that's faster, more reliable and tailored to specific needs.

4. Enhancing Coding Skills: Knowing how a computer manages memory and processes information allows you to write more efficient and optimized code, leading to better performance.

In summary, learning computer architecture gives the knowledge to make smarter choices when working with computers, whether it's coding, system design, or troubleshooting. It helps us understand how and why computers work the way they do.

** Layers of Computer:



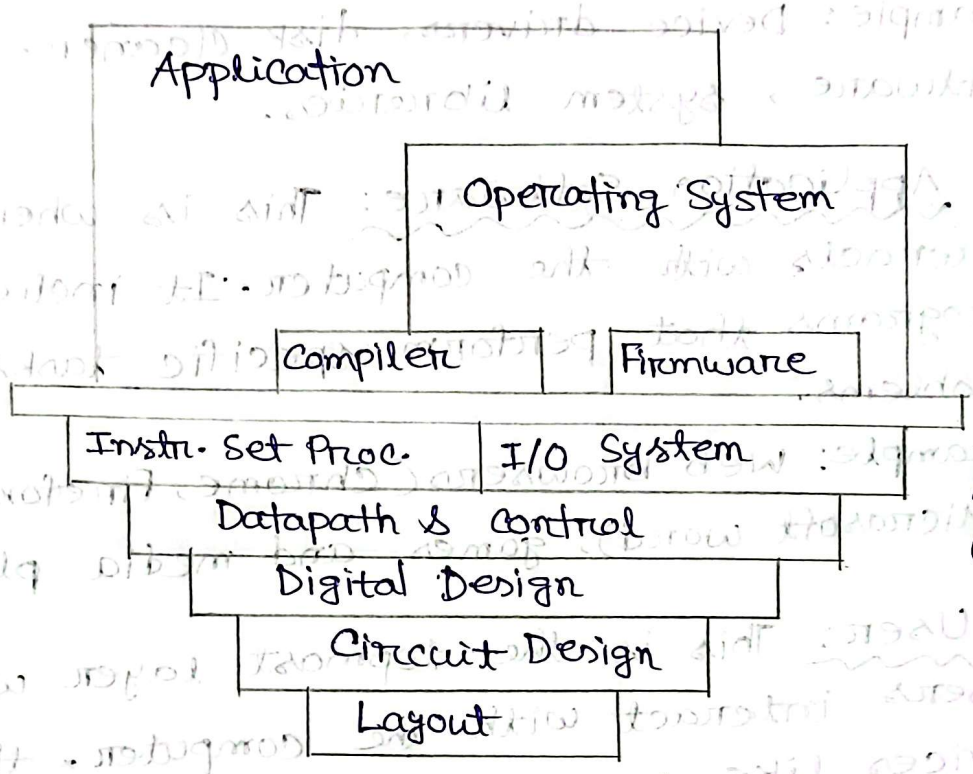
1. Hardware: This is the physical foundation of the computer. Ex
Examples: CPU, Memory, storage devices, motherboard, power supply etc.

2. System Software: Software that helps run and manage applications and hardware. It includes program that make the system work efficiently.
Example: Device drivers, disk cleaner, antivirus software, system libraries.

3. Application Software: This is where the end-user interacts with the computer. It includes the software programs that perform specific tasks or solve problems.
Example: Web browsers (Chrome, Firefox), word processors (Microsoft word), games and media player.

4. User: This is the topmost layer where human users interact with the computer through input devices like keyboard, mouse & touchscreens.
Example: The interface of applications (the desktop of an operating system, web browsers).

Levels of abstraction in computer architecture:



1. Application: This is the top layer. The programs we use, like games or web browsers.
2. Operating System: Manages the computer (Windows, macOS, Linux).
3. Compiler & Firmware:
   - Compiler turns programming code into machine language.
   - Firmware is special software that helps hardware work.
4. Instruction Set Architecture (ISA): This is the middle layer. The Define the set of instructions that the processor can execute.

5. Processor and I/O System:
- Instruction Set processing: Handles the execution of program instructions.
- I/O System: Manages input & output operations.

6. Datapath & Control: Moves data and controls how the processor works.

7. Digital ~~& Circuit~~ Design: Involves the logical design of circuits used in processors.

8. Circuit Design: Focuses on the electical design of transistors and logic gates.

9. Layout: This is the bottom layer. The physical design and placement of components on a chip.

~~for~~ Component of computer:

5 component of computer :-

(I) Control Unit
(II) Datapath
(III) Memory
(IV) Input
(V) Output

(I) Control unit:
- Directs how data flows inside the CPU
- Manages instructions from programs
- Act like a traffic controller for the computer.

(II) Datapath:
- Handles how data moves inside the processor
- Works with the control unit to process instructions.

(III) Memory:
- Stores data and instructions temporarily or permanently.
- Includes RAM (temporary memory) & ROM (permanent memory).

(IV) Input devices:
- Devices used to enter data into the computer.

Example: Keyboard, Mouse, Microphone, Scanner.

(V) Output devices:
- Devices that display or provide results after processing.

Example: Monitor, Printer, Speakers.

These components work together to process and executes tasks in a computer.