

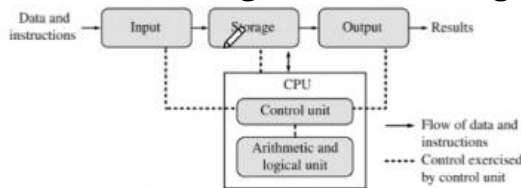
Unit I

1. What is a computer? With a neat diagram, discuss the organization of a computer (5) 2+1+2

Ans: A computer is an electronic device that is designed to accept data perform the required mathematical and logical operations and output the result. (data → process → information)

BASIC COMPUTER ORGANIZATION:

- Accepting data (input): keyboard mouse
- Storing data: primary and secondary memory
- Processing data: CU and ALU
- Displaying data (output): monitor printer
- Controlling and coordinating all operations inside a computer



2. Elaborate the different components of computer hardware

Ans: The different components of computer hardware are:

1. Memory:

- Internal storage area in the computer which is used to store data either temporarily or permanently.
- While the main or primary memory stores the data when a program is executing, the auxiliary or secondary memory stores data that are not currently in use and provides long term storage.
- Primary memory is volatile and can retain data only when power is on.
- RAM (Random Access Memory) and ROM (Read Only Memory) are two types of primary memory.
- RAM is volatile and stores temporary data.
- Types of RAM are
 - SRAM (Static RAM), holds data without an external refresh as long as it is powered.
 - DRAM (Dynamic RAM), contains millions of tiny cells each made of capacitor and transistor. If charge is present in the capacitor, the reading is 1 else it is 0.
- ROM is non-volatile and data is retained even after the computer is turned off.
- Types of ROM are PROM (programmable ROM), EPROM (Erasable programmable ROM) and EEPROM (Electrically EPROM). The process of writing an EEPROM is called flashing.

2. Disks:

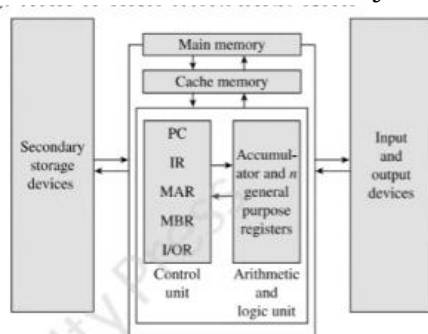
- Hard drive is a part of the computer that stores all the programs and files, so if in any case the drive is damaged, all the files are lost.
- Hard disk is a set of disks stacked together like phonographic records, that has data recorded electromagnetically in concentric circles called tracks.
- A single hard disk has several platters that are covered with a magnetic recording medium. Each platter required 2 R/W heads one for each side. This head can

pivot back and forth over the platters to read or write data on them. Data is stored on the surface of a platter in sectors and tracks.

- The performance of hard disk depends on its access time, which is the time required to read or write on the disk.
- Access time is a combination of 3 components:
 - Seek time: time taken to position the R/W head over the appropriate cylinder.
 - Rotational delay: time taken to bring the target sector to rotate under the R/W head.
 - Transfer time: time to transfer data or R/W to a disk.
- Access time = seek time + rotational delay + transfer time
- Disk latency (time taken to initiate a transfer) = seek time + rotational delay

3. Processor:

- It contains 2 main parts, ALU (Arithmetic and logical unit) and CU (Control unit). Besides these, there are also registers, execution unit, and bus interface unit.
- Execution unit = CU + ALU + registers
 - CU: to direct and coordinate the computer operations.
 - It interprets the instructions and initiates action accordingly.
 - It controls the flow of data and direct the others (alu, registers, i/o devices).
 - It is responsible for fetching, decoding, executing instructions and storing result.
 - ALU: performs arithmetic, comparison and other operations.
 - Stores all temporary results.
- Register is a computer memory that provides quick access to the data currently being used for processing.
 - Stores the final result.
 - Types of registers based on specific storage function include:
 - Accumulator and general-purpose registers: to store data brought from main memory n intermediate results.
 - Special purpose registers
 - Instruction cycle



4. Peripheral devices/ I/O devices

3. Write a note on Bus Interface Unit in detail

- Ans: BIU provides functions for transferring data between the execution unit of the CPU and the other components of the computer system that lie outside the CPU.
- Every computer system has 3 different types of buses to carry info from one part to another. They are control bus, data bus and address bus.
- BIU puts the contents of the program counter on the address bus.

- The content of the program counter is the next instruction to be executed.
- Once the memory receives an address from the BIU, it places the contents of that address on the data bus, which is then transferred to the IR of the processor through the MBR.
- Then the contents of the program counter are modified so that it now stores the next set of instructions.

4. Write a short note on Input/Output Devices

Ans:

- To accomplish tasks, a computer must be able to interact with its users. For this purpose, we need input and output devices.
- They are also called peripheral devices.
- Each input and output devices have its own function differentiating it from the others.
- i/p devices, used to feed data and instructions into the computer.
 - Keyboard, pointing devices, handheld devices, optical devices, audio/video input devices.
- o/p devices, gives info from the computer. They are electromechanical devices which accept digital data from computer and convert them to human understandable language.
 - Soft copy devices, produce electronic version of output. Monitor, projector, speaker.
 - Hard copy devices, produce a physical form of output. Printer.

5. Explain system software in detail

Ans:

- Represents programs that allow the hardware to run properly.
- It is transparent to the user and acts as an interface between the hardware of the computer and the application software that users need to run in the computer.
- Designed to operate the computer hardware and to provide and maintain a platform for running application software.
- Eg, BIOS (basic i/p and o/p system), key role is to load and start the operating system.
 - Its first function is to initialize and identify system devices.
 - Runs a series of tests called POST (power on self-test)
 - Locates software held on peripheral devices, loads and executes that software and gives it control of the computer (booting)
 - It is stored on a ROM chip built into the system and has a user interface like that of a menu that can be accessed by pressing a certain key on the keyboard when the computer starts.

6. Give the differences between application software and system software

Ans:

System software:

- Represents programs that allow the hardware to run properly.
- It is transparent to the user and acts as an interface between the hardware of the computer and the application software that users need to run in the computer

Application software:

- Computer software that employs the capabilities of a computer directly to perform user defined task.

- Designed to solve particular problems for the users
- Eg, spreadsheets, database systems, games, web browsers
- It represents programs that allow users to do something besides running that hardware.

7. What is an operating system? Give examples

Ans:

- Primary goal is to make the computer system convenient and effective to use.
- Offers generic services to support user applications.
- Human- computer interface which helps to identify and launch an application.

Utility software:

- It is used to analyse, configure, optimize and maintain the computer system.
- These programs may be requested by application programs during their execution for multiple purposes.
- Disk: increases efficiency.

8. List and explain the function of BIOS

Ans:

- BIOS (basic i/p and o/p system), key role is to load and start the operating system.
- Its first function is to initialize and identify system devices.
- Runs a series of tests called POST (power on self-test)
- Locates software held on peripheral devices, loads and executes that software and gives it control of the computer (booting)
- It is stored on a ROM chip built into the system and has a user interface like that of a menu that can be accessed by pressing a certain key on the keyboard when the computer starts.

9. Explain Compiler, Interpreter, Linker, and Loader

Ans:

- Compiler
 - Transforms the source code written in a programming language (source language) into machine learning language (target language)
 - The resultant code is in 0 and 1, object code (used to create an executable program)
 - Translate high-level programming language to a lower level language (assembly language / machine code)
 - If source code contains errors then the compiler will not be able to perform its tasks.
 - Syntax error: typing errors ; logical error: program errors
 - Its main work is to translate human readable source code to computer executable machine code.
 - Unless all error (if occurring) are fixed, source code cannot be converted to object code.
- Interpreter
 - Translates info into an intermediate form and then executes it
 - Line by line execution of source code
 - Compiled program executes faster than an interpreted one.
 - It immediately executes high level programs.

- Linker
 - Combines object modules to form an executable program
 - Usually a code is broken into smaller modules. When the source code of all the modules has been converted to the object code, all modules are put together using the linker.
 - Compiler usually involves the linker automatically as the last step in compiling a program
- Loader
 - Copies programs from a storage device to main memory to be executed.

10. With a neat diagram elaborate the process of design and implementation of efficient software programs

Ans:

- The design and development of correct, efficient and maintainable programs depends on the approach adopted by the programmer to perform various activities.
- o/p of one phase provides i/p for its subsequent phase.
- Requirement analysis: [WHY]
 - Expectations on why the program must be built.
 - Jot down the scope and objective of the overall software product
 - Documenting every identified requirement
- Design: [HOW]
 - A plan of actions is made
 - The core structure of the software/ program is broken down into modules
 - Specifies how the program will be built
- Implementation:
 - Designed algorithms are converted into program code using high level languages.
 - Construction or code generation phase
- Testing:
 - All modules are tested together to ensure that the overall system works well as a whole product
 - Integration testing
- Software deployment, training and support:
 - Installation and deployment in production environment
- Maintenance:
 - Enhancements on ongoing activities which are done to cope up with the newly discovered problems.

11. What are the different Program design tools available? Explain

12. What is an algorithm? What are its characteristics

13. Discuss the importance of flowchart in detail

14. What are the different shapes used in designing a flowchart? Give examples

15. Explain a pseudocode

16. Give the differences between pseudocode and algorithm

Ans:

- Algorithms:
 - Blueprint to writing a program to solve a particular problem.
 - Used to achieve software reuse.

- Characteristics of an algorithm include: precise, unambiguous, no repetition and result must be obtained.
- Sequence: executed in specified order
- Decision: outcome depends on some condition
- Repetition: executing one or more steps for a number of times. Loops.
- Flowcharts:
 - Graphical or symbolic representation of a process.
 - Helps users visualize the logic of the process.
 - Each step in the process is depicted by different symbols having different descriptions.
 - start/end: oval
 - i/o: parallelogram
 - processing/ steps: rectangle
 - condition: diamond
- Pseudocodes:
 - Compact and informal high-level description of an algorithm that uses the structural conventions of a programming language.
 - Mostly meant for human reading than machine reading.
 - Outline of a program that can be easily converted into programming statements.

17. Explain the different types of errors:

Ans:

- Run time errors:
 - occurs when the program performs some illegal operations (eg, dividing a number by zero)
 - may terminate program execution
 - the ability to continue operations of program despite of run time errors is called robustness.
- Compile time errors: occurs during compilation
 - Syntax errors: when rules of c prog are violated.
 - Semantic errors: may comply with the rule of c prog but are not meaningful to the compiler.
- Logical errors: errors in program that result in unexpected and undesirable o/p
- Linker errors: occur when the linker is not able to find the function definition for a given prototype.
-

18. Write an algorithm to find whether the given year is leap or not

Ans:

- Step1: start
- Step2: enter year as year
- Step3: check IF((year%4==0 and year%100 != 0) or (year%400 == 0)),
 Then PRINT "leap year"
 else:
 PRINT "not leap year"
- Step4: end

19. Draw a flowchart to find whether the person is eligible to Vote or not

20. Draw a flowchart to print the multiplication table of N (N should be an user input)

21. Write a pseudocode to swap two numbers