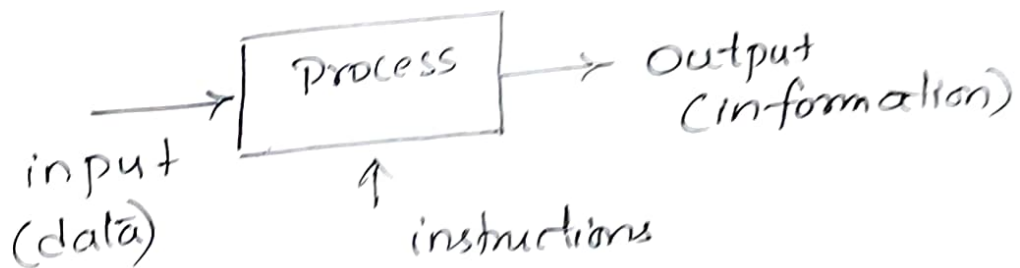


Unit - 1 programming for problem solving

Introduction to computers

Computer :- A computer is an electronic machine which accepts data and instructions as input, processes the data and produces meaningful information known as output, which can be represented as,

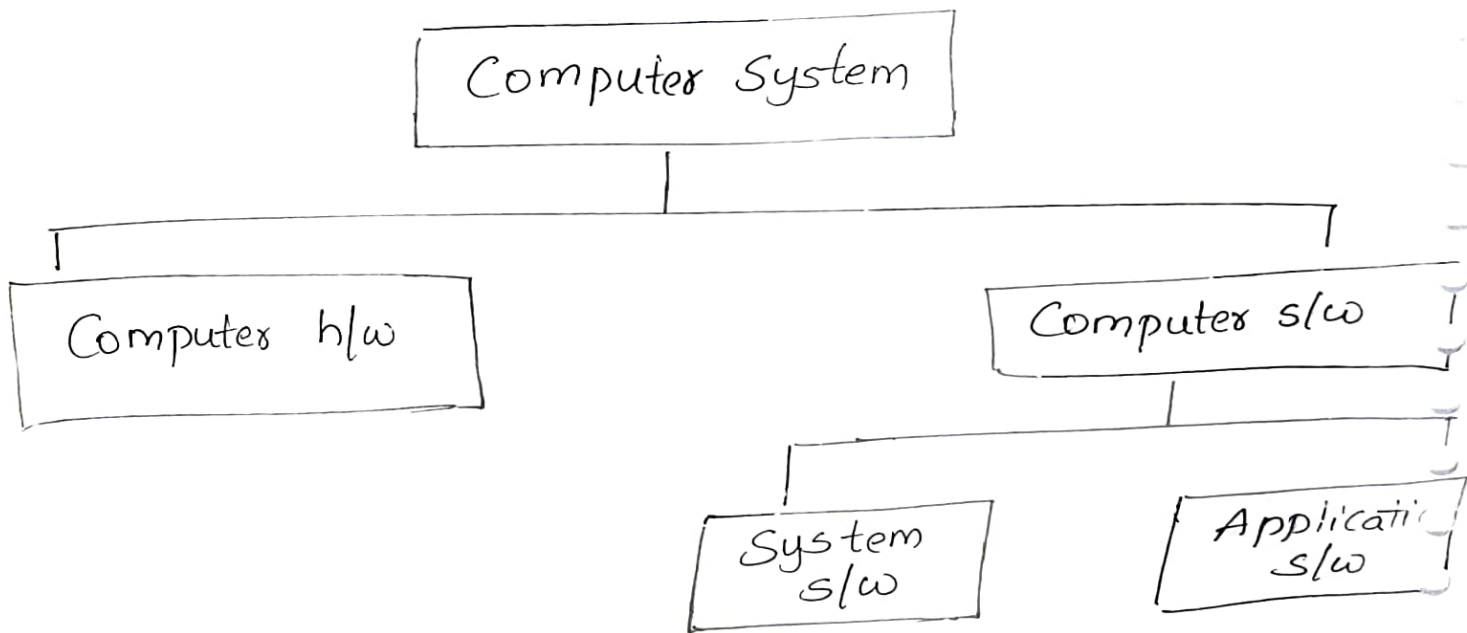


Data :- raw facts are called as data.
Eg: BE, Isem, CSE, online classes, 14 Dec, MJ COLLEGE, etc

Information :- Processed data is called as information.

Computer actually mean two things

- * Computer hardware (h/w) - which does all the physical work computers are known for
- * Computer Software (s/w) - commands the h/w what to do and how to do.



Computer Hardware

Part of machine, it can only understand two basic concepts - ON(1) and OFF(0) states, called binary.

Computer Software

the h/w needs a s/w to instruct what has to be done.

A program is a set of instructions that is arranged in a sequence to guide a computer to find a solution for a given task.

The process of writing a program is called programming.

Classification of Computer s/w

* System s/w & * Application s/w.

System s/w :

Provides a general programming environment in which programmers can create specific applications to suit their needs.

It acts as an interface b/w the h/w of the computer and the application s/w, i.e. System s/w is designed to operate the computer h/w to provide and maintain a platform for running application s/w.

Most widely used System s/w are Computer Bios and Device drivers which provide basic functionality to operate and control the h/w connected to or built in to computer.

BIOS - Basic Input/Output System - built into the computer and is the first code run by the computer when it is switched on.

The key role of the BIOS is to load and start the operating system. The first function that BIOS performs is to initialize and identify system devices like keyboard, mouse, harddisk and other h/w.

The code in the BIOS chip runs a series of tests called POST (Power on Self Test) to ensure that the system devices are working correctly.

BIOS is stored on a ROM chip built into computer.

Application s/w

Designed to solve a particular problem for users.

Programming Languages

The operation of a computer is controlled by a set of instructions (Program) which tells the computer.

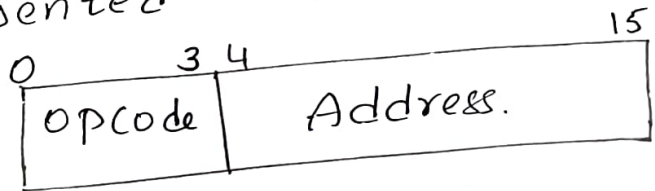
- *) what operation to perform
- *) where to locate data
- *) how to present results and make decisions.

The language used for ~~the~~ communicating with a computer is known as the programming language.

Programming languages are of 3 types as,

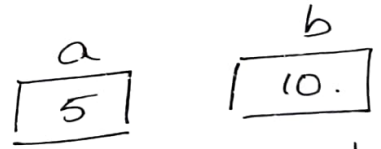
- * Machine language
 - * Assembly language
 - * High level language
- } low-level.
- [procedure oriented
 Problem oriented
 Natural lang.

Instruction:- is defined by using 16 bits (2 bytes) which consists of 2 fields as operation code (opcode) and Address part which can be represented as,

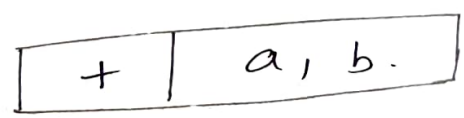


4 bits used to specify opcode part
12 bits to specify address.

Eg:- $C = a + b.$



to execute the above instruction, we need to fetch the values of a and b stored in memory at some address specified in address part of the instruction and then perform the operation specified in op code part of instruction.



Machine language:-

Only lang. that the computer understands.
All the commands and data values are expressed using 0's and 1's, corresponding to the OFF and ON electrical states in a computer.

The main advantage of m/c lang is that the code can run very fast and efficiently. Since it is directly executed by the processor.

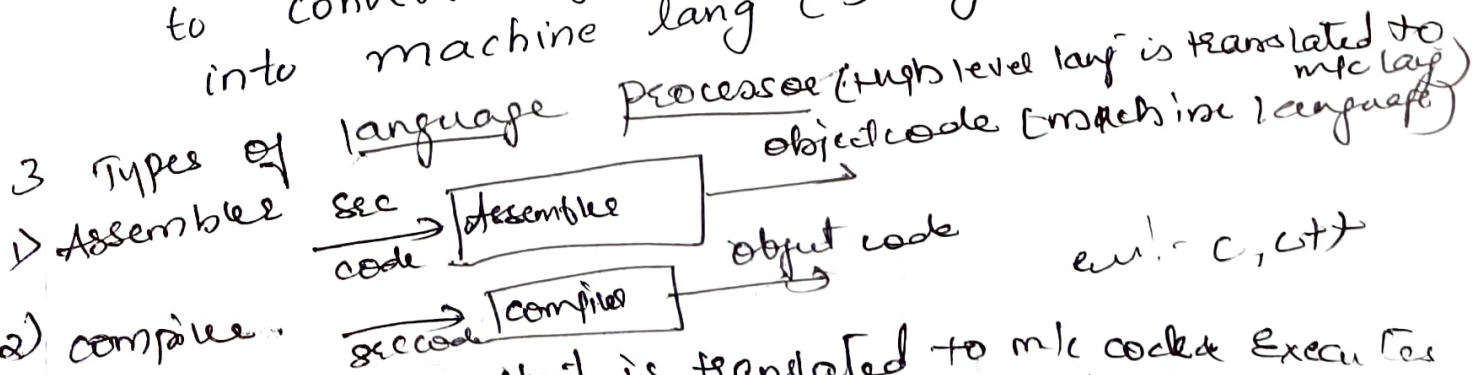
The m/c lang is difficult to learn and is far more difficult to edit if error occurs. The code written in m/c lang. is not portable across systems.

Assembly language

It uses symbolic notation to represent machine instructions. The symbolic codes used are known as mnemonics and are easy to remember abbreviations rather than numbers in binary.

Eg:- ADD, SUB, MUL, DIV, CMP, LOAD, STORE etc.

The code written in assembly lang. with needs a translator known as the assembler to convert symbolic constants used in instructions into machine lang (binary).



3) interpreter - Single stmt is translated to m/c code & executes immediately before moving on to next line (only after removing error).

High level languages

(4).

In this programs are written in English-like manner, making them more convenient to use and giving the programmer more time to address a problem.

A translator is required to translate the instructions written in high-level lang. into computer executable m/c lang, known as Compiler (or) Interpreter.

These are easy to write and debug as well as they are portable across m/c's.

Procedure oriented lang

It follows a step by-step approach to solve a task through a sequence of instructions.

Eg:- C

Problem-Oriented lang

no standard rule to define, they are non-procedural, concentrate on what instead of how. Mostly code is written in English-like sentences.

- e.g.

Eg:- SQL - query lang.

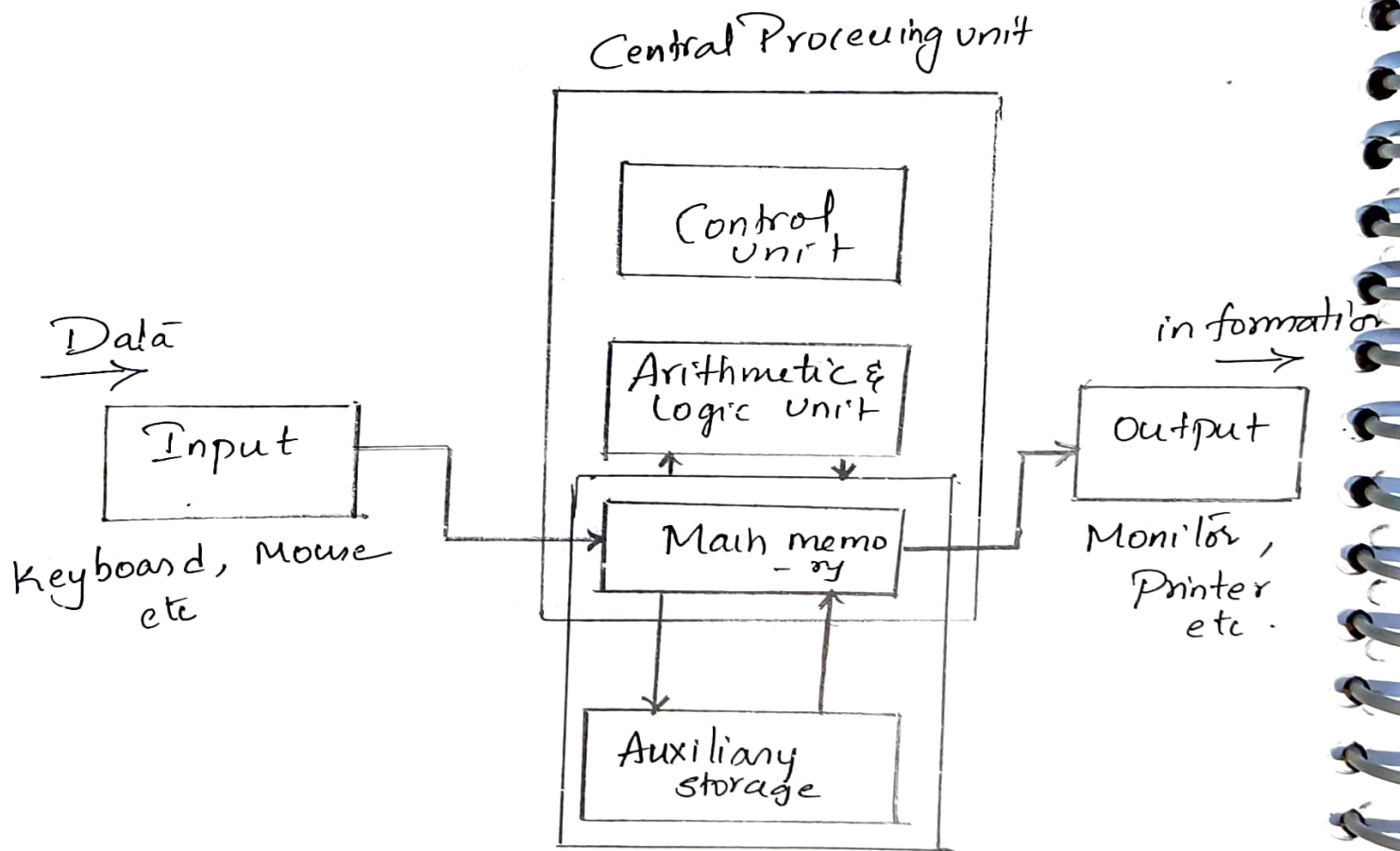
Natural lang

Centred on solving problems using constraints given to the program.

Eg:- Robotics, Neural N/w's.

Block diagram of a Computer

The working process of a computer from inputting the data to retrieving the desired results can be represented using block diagram as,



Memory Unit

Input Unit :- consists of input devices which are used to input information (or) instructions into the computer system.

Two major functions of input unit are:-

- * It converts the inputted data (or) instructions into binary form for further processing.
- * It transmits the data to the main memory of the computer.

Central Processing Unit (CPU)

CPU is known as the brain of the computer. It is an electronic h/w device that processes all the operations (arithmetic and logical). All the calculations, operations are performed inside the CPU.

The control unit (CU) and Arithmetic & Logic Unit (ALU) are jointly called the CPU.

The control unit of a CPU controls all the activities and operations of the computer. It is also responsible for controlling Input/output, memory and other devices connected to the CPU.

The control unit determines the sequence of operations to execute the given instructions.

The ALU performs arithmetic and logical operations. Apart from these, it also performs several other operations such as comparing, selecting, matching and merging data.

The information or data is transmitted to ALU from the storage unit only when it is required.

Memory Unit :

used to store data and instructions before and after processing. The memory unit transmits the information to other units of the computer system when required.

We have two types of memory units as Primary memory and secondary memory.

Primary Memory (Main / Built in)

Available as a built in unit of a computer. The primary memory is represented as a set of locations with each location occupying exactly 8 bits (1 byte). Each byte in the memory is identified by a unique address and the data is stored in the binary form in these memory locations.

Commonly used primary memories are :-

ROM (Read only memory) :-

Stores data and instructions even when the computer is turned off. It is the permanent memory of the computer where the contents cannot be modified by user.

ROM is a chip which is inserted into motherboard (used for connecting CPU with the Input/output devices). ROM is generally used to store the Basic Input/Output system called as BIOS which performs POST.

RAM (Random Access Memory) :-

(6)

RAM is the read/write unit in which the information is retained only as long as there is power supply. The data will be lost if they are disconnected from the power supply.

RAM is volatile memory that temporarily stores data and applications as long as they are in use.

Secondary Memory (Auxiliary/Permanent memory)

It is not possible to store data permanently for future access using primary memory, so to have secondary memory.

Eg:- Hard disk.

Note:- Primary memory is the only memory that is directly accessible to the CPU.

Output Unit

It consists of devices, that are used to display the results or output of processing.

Eg:- Monitor, Printer etc.

It accepts the data or information in binary form from the main memory and converts the data into human readable form.

Algorithm and Flow chart

A typical programming task can be divided into two phases:

* Problem Solving Phase :- In this phase, we produce an ordered sequence of steps that describe solution of a problem, known as algorithm. i.e. algorithm is a set of well defined instructions in sequence to solve the problem.

* Implementation phase - Implement in some language.
Characteristics / Feature - Algorithm.

- * Input and output should be defined precisely.
- * Each step in algorithm should be clear and unambiguous.

Eg:- $C = a + b$, $C = a - b \rightarrow$ no clarity to store what answer inc.

* Algorithm should be effective and should complete in a finite time.

* Algorithm shouldn't have computer code.

Eg:- To add 2 nos.

Step 1: Start

Step 2: Declare 2 vars n_1, n_2 and sum

Step 3: Read values for n_1 and n_2

Step 4: Add n_1 and n_2 and store the result in sum
 $Sum = n_1 + n_2$

Step 5: Display sum

Step 6: Stop.

find max among 2 nos.

(7)

Step 1: start

Step 2: Declare 2 vars a and b

Step 3: read values for a and b

Step 4: check if $a > b$

display a is max no.

else display b is max no.

Step 5: stop.

Advantages of algorithm

1. As it is a step-wise representation of a solution, makes it easy to understand.
2. Uses a definite procedure.
3. Easy to debug - follows logical sequence

Disadvantage

- *) takes time
- *) algorithm is not a computer program.

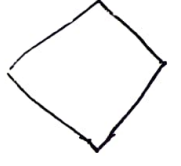
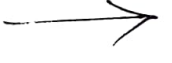
Flow Chart

A flowchart is a graphical representation of workflow or process. It shows steps in sequential order and is widely used in presenting the flow of algorithms, workflow or processes.

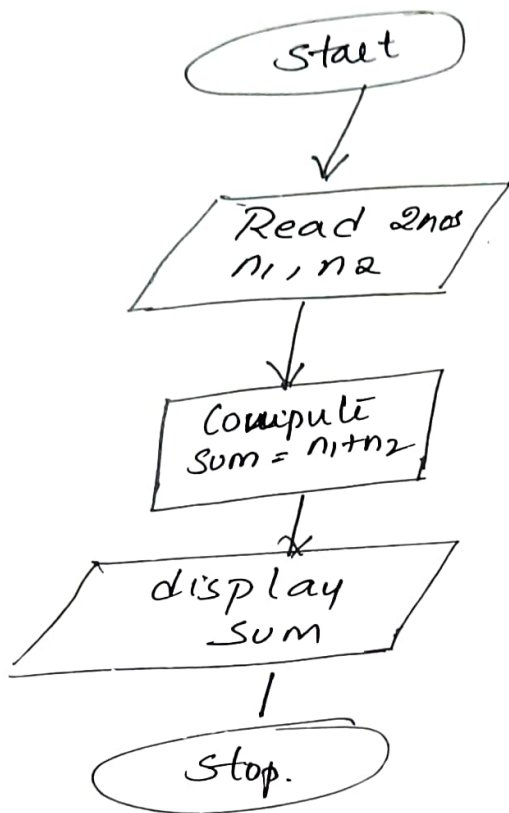
Note: Pseudocode :- Informal lang. which helps programmer develop algorithm.

A flow chart is a diagrammatic representation of an algorithm. A flow chart can be helpful for both writing programs and explaining the program.

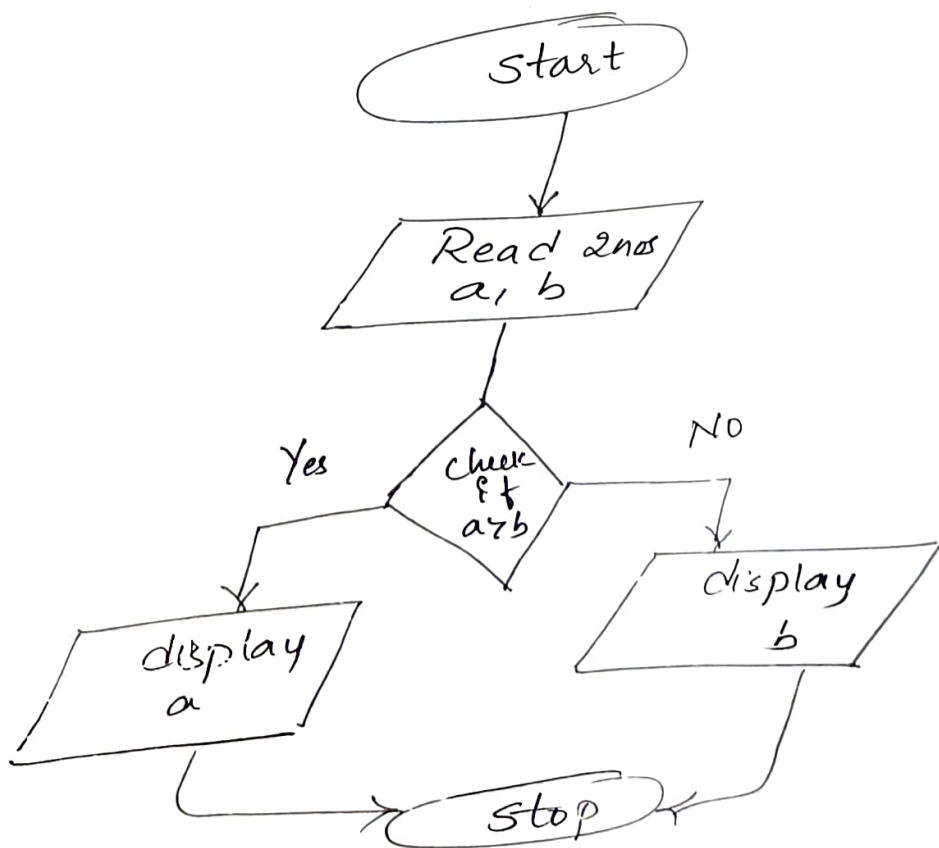
Symbols used in flow chart

<u>Symbol</u>	<u>Purpose</u>	<u>Description</u>
	terminal (start/stop)	represents the start and the end of a flow chart
	Input /out	used for Input/output operation
	Processing	used for arithmetic & data manipulation
	Decision	used for decision making b/w two or more alternatives
	flowline	Indicates the flow of logic by connecting symbols
	Connector	used to join different flowlines.

flow chart to add 2nos



flow chart to find max among 2nos.



Advantages of flow chart :-

- *) Convenient method of communication
- *) A key to correct programming
- *) Indicates the role-played at each level
- *) facilitates troubleshooting
- *) Promotes logical accuracy.

Disadvantages of flow chart

- *) waste of time and slows down the process of s/w development.
- *) Sometimes the complex logic of the program is quite complicated to draw out.
- *) Modification or alternation of the process is very hard.

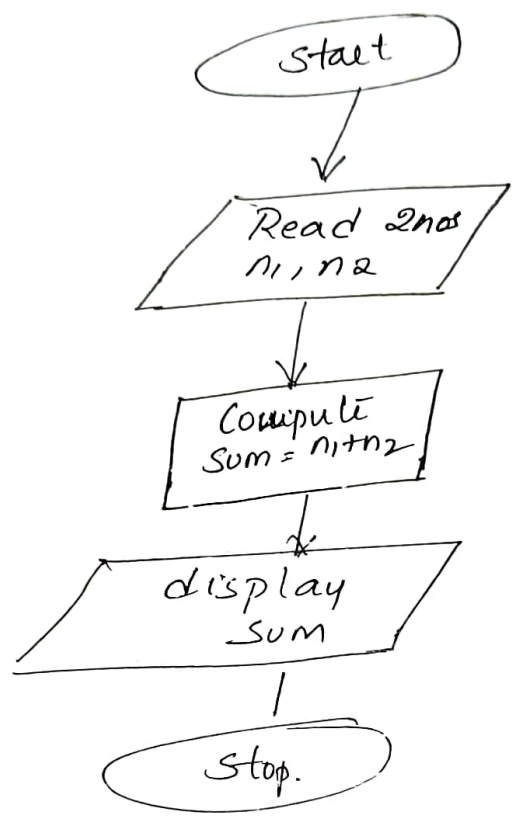
Computing Environments

It's a collection of computers which are used to process and exchange information to solve various types of computing problems.

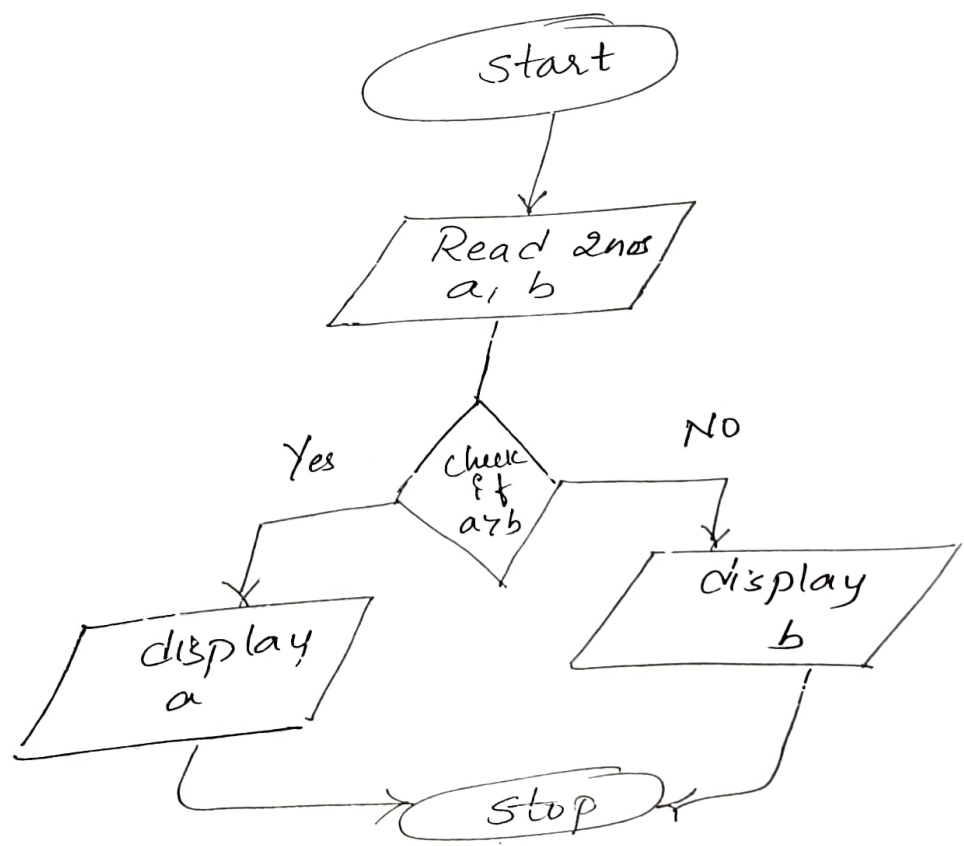
Types of Computing Environments :-

1. Personal Computing Environment
2. Time-sharing computing Environment
3. Client-Server Computing Environment
4. Distributed computing Environment
5. Grid Computing Environment
6. Cluster Computing Environment.

flow chart to add 2nos



flow chart to find max among 2nos.



Advantages of flow chart :-

- *) Convenient method of communication
- *) A key to correct programming
- *) Indicates the role-played at each level
- *) facilitates troubleshooting
- *) Promotes logical accuracy.

Disadvantages of flow chart

- *) waste of time/and slows down the process of s/w development.
- *) Sometimes the complex logic of the program is quite complicated to draw out.
- *) Modification or alteration of the process is very hard.

Computing Environments

It is a collection of computers which are used to process and exchange information to solve various types of computing problems.

Types of Computing Environments :-

1. Personal Computing Environment
2. Time-sharing Computing Environment
3. Client-Server Computing Environment
4. Distributed Computing Environment
5. Grid Computing Environment
6. Cluster Computing Environment.

Personal Computing

It is a stand-alone machine. The complete program resides on stand-alone machine and executed from the same machine.

Eg:- Laptops, printers, mobile devices, computer syst at home.

Time-sharing Computing

It is a stand-alone computer in which a single-user can perform multiple operations at a time by using a multitasking operating system. The processor time is divided among different tasks which is called Time-sharing.

Eg:- We can listen to music while writing some thing in a text editor.

Client-Server Computing

It contains two machines (Client & Server). They both exchange the information through an application like PC, Tablet, Mobile etc. Client is normal computer and server is a powerful computer which stores huge data and manages the huge amount of file.



Distributed Computing

The complete functionality of the sw is not on a single computer but is distributed among multiple computers. These computers communicate with each other over a n/w to perform the complete task. It is a system with multiple components located on diffn m/c, that communicate & coordinate actions to appear as a single coherent system to End user.

Grid Computing

It is a collection of computers from different locations, which work for a common problem.

Cluster Computing

It is a collection of interconnected computers which work together to solve a single problem.

Program Development Life Cycle

The set of steps or phases used to develop a program in any programming lang, and contains the following phases:-

- * Problem definition - define the problem stmt, what is our requirement, what should be the output of the problem.
- * Problem Analysis - the requirements to solve the problem - gathering the required resources to solve the problem.
- * Algorithm Development - develop step-by-step procedure to solve the problem.
- * Coding and Documentation - to write or implement the actual programming instructions for steps defined in algorithm.
- * Testing & Debugging - check whether the code written is solving the specified problem or not.
- * Maintenance :- the program is actively used by the users.

Language Processors

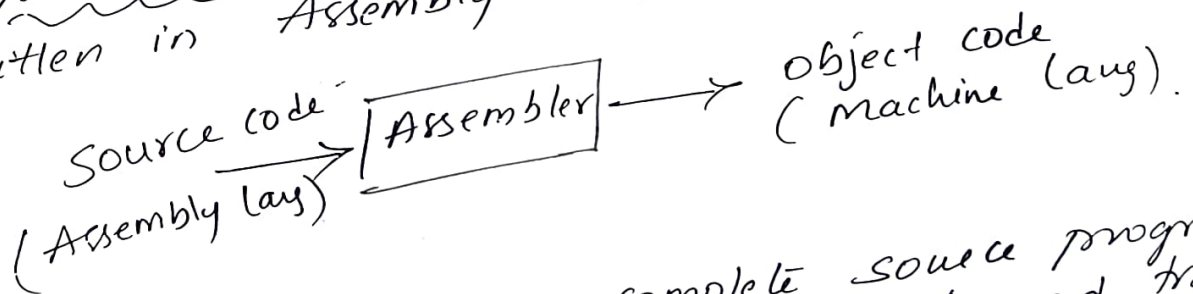
A computer understands instructions in machine code i.e. in form of 0s and 1s.

The source code cannot be executed directly by the computer and must be converted into machine lang. to be executed.

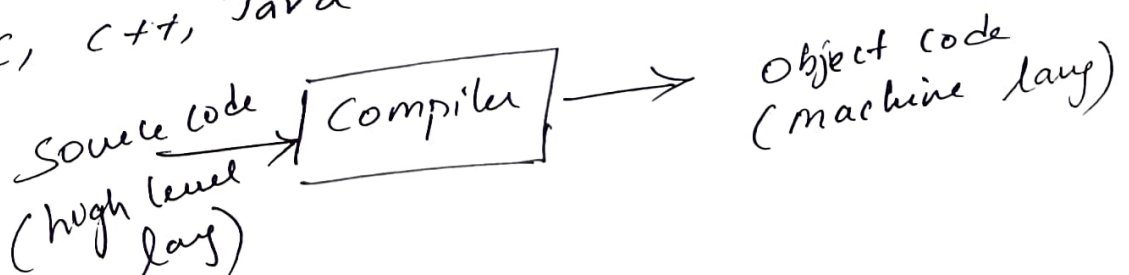
A special translator system software is used to translate the program written in high-level lang. into machine code is called lang. processor and the program after translated into machine code is called object code.

three types of lang. processors are:-

1) Assembler:- used to translate the program written in Assembly lang. into machine code.



2) Compiler:- reads the complete source program written in high level lang. as a whole and translates it into machine lang.
Eg:- C, C++, Java



Interpreter:- the translation of single stmt of source program into machine code is done by and executes it immediately before moving on to the next line is called interpreter.

The interpreter moves on to the next line only after removal of the error.

Eg:- Python, Matlab.

Compiler

- 1) Scans the entire program and translates it as a whole into machine code
- 2) the overall execution time is comparatively faster than interpreter.
- 3) Generates intermediate code which further requires linking, requires more memory.

Eg:- C, C++, Java etc

Interpreter

Translates program one stmt at a time

the overall time of execution is slower than compiler.

No intermediate code is generated, memory efficient

Eg:- Python, Javascript, Matlab etc.