Computational Problem Solving

06 September 2023 08:03

LIMITS OF COMPUTATIONAL PROBLEM SOLVING

- Complexity
- Efficiency
- Available resources

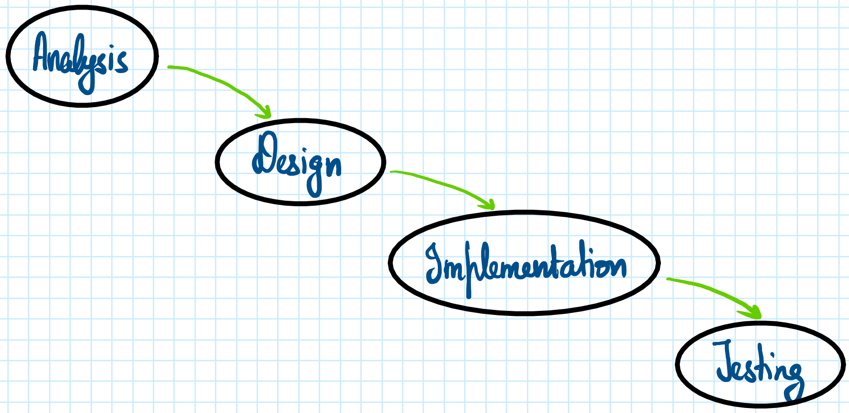
ESSENCE OF COMPUTATIONAL PROBLEM SOLVING

- Represent relevant information
- Algorithm for solving the problem

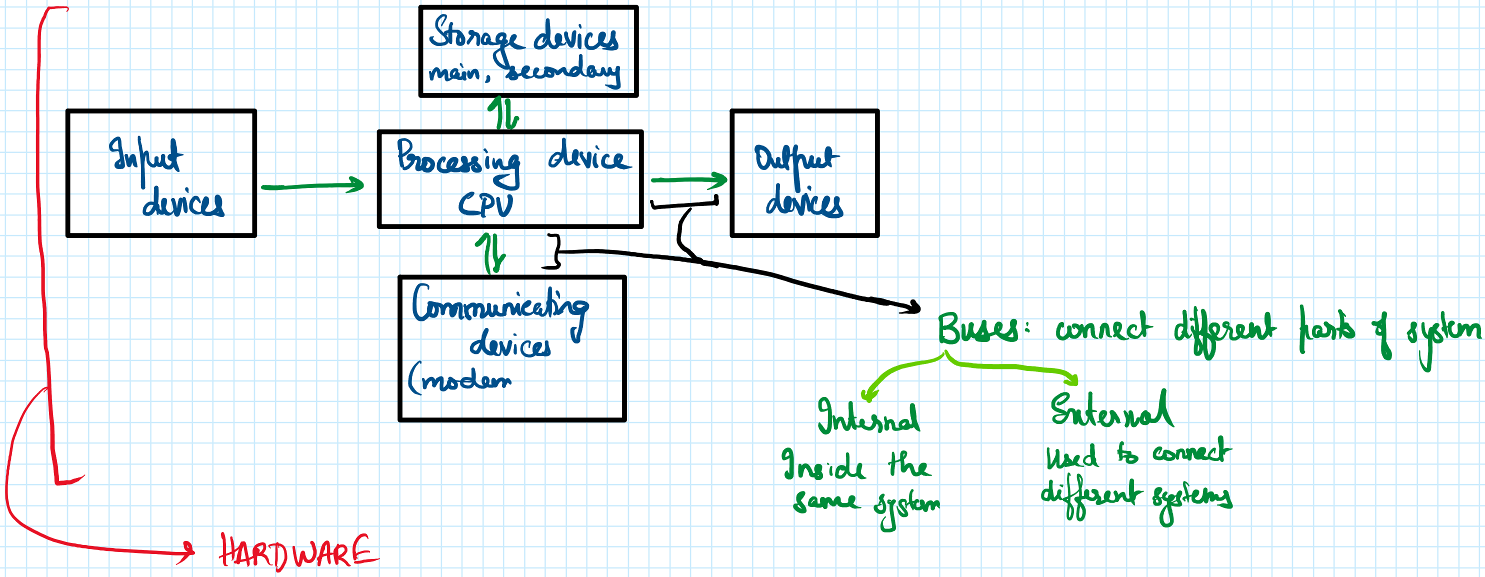
Algorithm → set of unambiguous steps

> NOTE : Brute force
> Considering every possible case or situation for representation regardless of validity.

PROCESS OF CPS



COMPUTER : A BASIC OVERVIEW



SOFTWARE → Application software

SOFTWARE → System software

NUMBER SYSTEMS

Binary	Decimal	Octal	Hexadecimal
0 & 1	0-9	0-7	0-9, A-F
Base : 2	Base : 10	Base : 8	Base : 16

CONVERSIONS

[something] to decimal :

$\overset{2}{x}\overset{1}{y}\overset{0}{z}_{(b)}$

Decimal value = $b^2x + b^1y + b^0z$

Decimal to [something] :

$xyz_{(10)}$

Divide by base of [something] until you get a quotient less than the base.
Note the remainders in every step.
Reading remainders in reverse order gives the converted value.

LEVELS OF PROGRAMMING LANGUAGES

Low level

- Directly understood by machine
- Programmer resposibility is more

Eg: Machine language

Middle level

- Basic data structure and array definition
- Programmer should take care of operations
- Uses compiler

Eg: C, C++

High level

- Programmer concentrates on algorithm and programming

Eg: Python

- Programmer concentrates on algorithm and programming

Eg: Python

PARADIGMS OF PLs

Imperative

- do this, then do this, then do this

Eg: Fortran, COBOL, Pascal

Procedural

Declarative

Functional

Logical

Object-Oriented

TRANSLATORS

Compiler

- converts entire high-level language to machine

Interpreter

- converts high-level language line by line

- converts entire high-level language to machine all at once
- does not execute; CPU executes
- one error → no output

- converts high-level language line by line
- executes the code instead of CPU. This is done line by line
- one error → all output of lines before the error