UNIT - 3

LISP :-

LISP means -" List-Processing". LISP is one of the oldest Computer - Programming Language.

It was invented by "John Mccarthy" during the Late 1950's, Shortly after the development of FORTRON.

LISP is Suitable for AI programs because of it has ability to process Symbolic information effectively. Special LISP Processing machines have been built & it's popular to many new Sectors of business & government.

Characteristics of LISP →

LISP Consists following

features —

(I) Simple - Syntax
(II) with little or no data - typing
(III) Dynamic Memory Mgmt.

Syntax & Numeric Functions →

The basic building blocks of LISP are follows —

(I) atom   (II) List   (III) String

atom →

An atom is a number or string of contiguous- characters. it includes underscore(_) Special- Character.

---

for example →

Valid Atoms	Invalid atoms
Bee_Systems	(abc
BSM	123a
Hello	abc'ef
a123	(ab)
block #6	ab cd

2) List :-

A List is a Sequence of atoms and/or other lists enclosed within Parenthesis.

for example →

Valid_List	Invalid_List
(BSM Bee-Sims)	(abc
()	abc)
(a (a b))	abc
(Mon tue wed thus frid Sat)	)abc(

3) String →

A String is a group of characters, enclosed in double quotation marks.

for example →

Valid String	Invalid String
"This is a string"	"This is not a String
"abp A B # s y."	nor" this"
"Bee System"	this"

Some other Definition →

i) Top-element → The element whish is Consist by List, Called Top-elements.

for example → (a b (c d) e (f) is a list
The Top-element of above -list are —

   a, b, (c,d), e, (f)

and Top-element of sub-list (c,d) is c, d

2). Symbolic-expression or S-expressions →

Atom, lists and Strings are the only valid objects in LISP. They are called Symbolic-expression or S-expressions.

→ LISP Programs run either on an interpreter or as Compiled code.

⇒ The interpreter examines Source Programs in a repeated loop, called the "Read-evaluate Print loop". This loop reads the Program code, evaluates it and Prints the Values returned by the Program. '→' Symbol is used as a Prompt.

for example → To find the Sum of the three nos. 5, 6 & 9, we type after the Prompt.
we can write

   →(+ 5 6 9)
      20
   →

LISP Function:-

1) Predefined Numerical function →

The basic numeric operations are +, -, * and /. Arguments may be real or integers values and function takes different no. of arguments.

for example ⟹
+ & * normally takes two or more arguments while - & / take two arguments Nil is also the Same as the empty-list.

Predefined Numeric Functions →

function. Call	Valued Return	Remarks
(i) (+ 35 -84)	20	+ takes zero or more arguments. The Sum of zero argument is 0.
(ii) (- 10 12)	2	- takes two arguments
(iii) (* 2 3 4)	24	* takes zero or more arguments
(iv) (/ 25 2)	12.5	/ takes two arguments