Q1. Study about LISP programming, Applications, Different versions, Operations, Programming environment, Platforms etc.

LISP Programming -

Lisp i.e., list processing is a programming language that has an overall style that is organized around expressions and functions. Every Lisp procedure is a function, and when called, it returns a data object as its value. It is also commonly referred to as "functions" even though they may have side effects.

Applications Built in LISP -

Large successful applications built in Lisp.

- Emacs
- G2
- AutoCAD
- Igor Engraver
- Yahoo Store

Different Versions of LISP -

- LISP 1
- LISP 1.5
- Stanford LISP
- MACLISP
- InterLisp
- Franz Lisp
- XLISP
- PSL (Portable Standard Lisp)
- ZetaLisp
- LeLisp
- Scheme
- CL (Common Lisp)
- Dylan
- EuLisp
- SBCL (Steel Bank Common Lisp)
- ISLISP
- ANSI Common Lisp
- ACL2

- Clojure
- GOAL (Game Oriented Assembly Lisp)
- Arc
- LFE (Lisp Flavoured Erlang)
- Nu
- Hy

Operations of LISP Programming –

- Arithmetic Operations
- Comparison Operations
- Logical Operations
- Bitwise Operations

Connection to artificial intelligence -

Since inception, Lisp was closely connected with the artificial intelligence research community, especially on systems. Lisp was used as the implementation of the language Micro Planner, which was used in the famous AI system SHRDLU. In the 1970s, as AI research spawned commercial offshoots, the performance of existing Lisp systems became a growing issue, as programmers needed to be familiar with the performance ramifications of the various techniques and choices involved in the implementation of Lisp.

Platforms –

LISP is supported on various Cisco platforms, but primarily the Cisco platforms are characterized based on the operating system software running on them. Cisco platforms support LISP on IOS/IOS-XE, IOS-XR, and NX-OS software.

Features of LISP Programming Language:

- It is a machine-independent language
- It uses iterative design methodology and is easily extensible
- It allows us to create and update the programs and applications dynamically.
- It provides high-level debugging.
- It supports object-oriented programming.
- It supports all kinds of data types like objects, structures, lists, vectors, adjustable arrays, set, trees, hash-tables, and symbols.
- It is an expression-based language
- It can support different decision-making statements like if, when, case, and condition.
- It will also support different iterating statements like do, loop, loop for, do times and do list.
- It will support input and output functions
- By using lisp, we can also create our own functions

Q2. Study about PROLOG programming, Applications, Different versions, Operations, Programming environment, Platforms, Advantages etc.

PROLOG Programming -

Prolog is a logic programming language. Any program written in a logic programming language is a set of sentences in logical form, expressing facts and it has important role in artificial intelligence. Unlike many other programming languages, Prolog is intended primarily as a declarative programming language rule about some problem domain. Prolog is intended primarily as a declarative programming language: the program logic is expressed in terms of relations, represented as facts and rules.

Applications of Prolog -

1. Intelligent Database Retrieval.
2. Natural Language Understanding.
3. Specification Language.
4. Robot Planning.
5. Automation System.
6. Problem Solving.
7. Pattern matching over natural language parse trees.

Different Versions of Prolog –

ISO Prolog
There are three corrigenda:
* Cor. 1:2007
* Cor. 2:2012
* Cor. 3:2017

Operations of PROLOG Programming -
* Arithmetic Operators
* Comparison Operators

Platforms –

Prolog is a logic programming language associated with artificial intelligence and computational linguistics. B-Prolog, Ciao, Eclipse, GNU Prolog, Poplog Prolog, P#, Quintus Prolog, SICStus, Strawberry, SWI-Prolog, Tau Prolog, tuProlog, WIN-PROLOG, XSB, YAP.

Features of Prolog Programming Language:
1. Unification: The basic idea is, can the given terms be made to represent the same structure.

2. Backtracking: When a task fails, prolog traces backwards and tries to satisfy previous task.

3. Recursion: Recursion is the basis for any search in program.

4.   Running queries

Advantages:

1. Easy to build database. Doesn't need a lot of programming effort.

2. Pattern matching is easy. Search is recursion based.

3. It has built in list handling. Makes it easier to play with any algorithm involving lists.