

## Proyecto no.1 -Algoritmos y Estructura de Datos:

### Lenguaje de Programación Lisp

- **Historia:**

En 1958, nació el sueño de John McCarthy en crear un lenguaje de programación el cual sería el estándar en la industria, desde el nivel principiante hasta el nivel del Departamento de Defensa de Estados Unidos. El lenguaje tuvo su comienzo con Lisp1; este fue visualizado como la clave para el desarrollo de inteligencia artificial, sin embargo, presentaba muchos errores de lógica. En 1962 el esfuerzo de McCarthy por arreglar los fallos de Lisp1 culminaron en la publicación de Lisp1.5; como se planeaba la publicación de Lisp2 en posteriores años se hizo el mínimo esfuerzo por corregir las características de Lisp en la versión 1.5, sin embargo, el financiamiento de investigación por parte de Massachusetts Institute of Technology (MIT) termino abruptamente, cancelando la publicación de la versión 2 de Lisp. (KABUTOKU CHANNEL, 2023)

Lisp se vendió a los sectores industriales como la clave de la innovación, por ello muchas empresas continuaron con su propio desarrollo del lenguaje de programación Lisp, esto culminó en un gran desastre. En 1965 a 1979 aparecen múltiples versiones del lenguaje Lisp; como en el caso de MaCLisp, Interlisp y Lisp Machine; todos ellos derivan de la versión 1.5 de Lisp, pero no todos continuaron la misma ruta de pensamiento. A finales de los 70 el futuro de Lisp era incierto por la incompatibilidad de las múltiples versiones de Lisp, específicamente en el apartado de las librerías, el recorte en el presupuesto de investigación de inteligencia artificial (AI Winter) y el poco progreso logrado respecto a desarrollo de IAs; por ello en 1980 a 1990 comienza el esfuerzo de estandarización de Lisp. En 1981 se reportan los primeros logros de estandarización, resultando en la creación de Common-Lisp, este incorporaba todos los aspectos positivos de las versiones alternas de Lisp al dar compatibilidad con la mayoría de las librerías creadas. En 1984 se publica el libro, Common Lisp The Language por Steele Guy L, el cual recopila las sugerencias y estándares a tomar en cuenta para la creación de programas en Lisp, este no fue considerado como la versión oficial definitiva debido a la simplificación en ciertos aspectos. En 1986 empieza el proyecto Anti Common Lisp, este busca la simplificación a su máximo nivel de Common Lisp para mejorar el rendimiento y tiempos de ejecución del lenguaje, este no fue visualizado para futuras actualizaciones. En 1990 Steele Guy publica la segunda edición de Common Lisp The Language. En 1994 finaliza el proyecto Anti Common Lisp y lo vuelven accesible a un mayor público, sin embargo, este proyecto llegó demasiado tarde.

Mientras se daba la estandarización de Lisp fueron naciendo otros programas los cuales tuvieron una mejor visión en el aspecto de la compatibilidad entre librerías, programas como C, Pearl, PHP y java, dichos programas le quitaron el puesto a Lisp como la opción preferida de la industria para generar avances tecnológicos. La popularidad de Lisp en la comunidad cayo de forma abismal, el poco uso que le daban era destinado a la enseñanza de los paradigmas de programación básicas en el MIT, esto por la simplicidad que adquirió el lenguaje en su versión Lisp Scheme, pero fue abandonado por la preferencia de utilizar nuevos lenguajes como el caso de Phytton. Hoy en día hay muchos programadores que no saben de la existencia de Lisp, por la falta de actualizaciones y de popularidad, sin embargo, es un lenguaje de programación muy completo por las múltiples iteraciones que tuvo al pasar de los años. (KABUTOKU CHANNEL, 2023)

- **Características propias:**

Lisp utiliza como base un paradigma de programación totalmente diferente al utilizado en Java y Phytton, a este paradigma se le denomina funcional. Descubierta en 1930, el paradigma de programación funcional busca solucionar el problema sin alterar el estado o características de los datos ingresados, asegurando que el resultado que retorna depende únicamente de los datos previamente ingresados. Al enfocarse en dicho paradigma, Lisp facilita la realización de cálculos simples mediante la manipulación de símbolos y procesamiento de listas. En casos como Java el proceso para ejecutar un código es lineal con tres pasos, escribir código – copilar el código – correr el código, mientras que en Lisp el orden de ejecución es un ciclo continuo, correr el programa – escribir el programa – copilar el programa, esto se logra mediante el proceso Lisp Image. El proceso Lisp image consiste en la analogía de cargar un lienzo en blanco y mientras se va agregando detalles el lienzo se va actualizando, en otras palabras, se carga una versión previa del programa el cual va cambiando acorde al código ingresado. Debido a la técnica de ejecución de código de Lisp Image, Lisp se vuelve una excelente herramienta de prototipado al incorporar una versión diferente por cada cambio ingresado, por ello se puede decir que Lisp es para el código, lo que es React Native para el diseño web. Actualmente se utiliza tres versiones diferentes de Lisp siendo: Lisp Scheme, Lisp Clojure y Common Lisp.

Existen diferentes comandos para simplificar el proceso de programación dentro de Common-Lisp, algunos de estos comandos incluyen:

- Setq & defparameter: les asignan un valor a las variables previamente instanciadas.
- Print & Format: permiten la entrada de datos ingresados por el usuario y mostrar el resultado de dichos datos.
- Car, cdr, cons & append: operar los datos ingresados dentro de una lista.
- Defmacro: modifica partes del código previo a ser copilado.
- If, cond & case: inicia los bucles condicionales dentro del programa.
- Loop, dotimes & dolist: inicia los bucles de rango de datos dentro del programa.

(Boticario J.G. 1996; KABUTOKU CHANNEL, 2023)

- **Conclusión:**

Lisp es mas que un lenguaje de programación olvidado en el pasado, es una oportunidad para experimentar el paradigma funcional, es una herramienta de actualización continua en el proceso de creación de un programa, es un facilitador para entender la lógica principal detrás de la programación, es un programa que busca resolver enigmas mediante el procesamiento de listas sin alterar el estado de sus contenidos y es un recordatorio de la importancia de implementar estándares o características definidas al diseñar proyectos o herramientas de acceso general. Es una pena que pocas personas tengan conocimiento del lenguaje Lisp debido a todas sus funciones que simplifican el proceso de entendimiento de código y por el hecho que el sueño de John McCarthy no se cumplió al 100%, por los múltiples inconvenientes en el camino, sin embargo, tras la salida de múltiples versiones de Lisp y creación de estándares que regulan dichas versiones resultaron en un lenguaje de programación muy completo y sencillo de entender.

**Referencias:**

- Boticario, J. G. (1996). *Introducción al lenguaje COMMON LISP*. Universidad Nacional de Educación a Distancia.
- KABUTOKU CHANNEL. (2023, septiembre 10). *The Rise & Fall of LISP - Too Good For The Rest Of the World*  
YouTube. <https://youtu.be/GVyoCh2chEs>
- KABUTOKU CHANNEL. (2023, septiembre 12). *Why Lisp Is One Of The Most Productive Programming Languages*  
YouTube. <https://youtu.be/GVyoCh2chEs>