

La era de las cavernas digitales: Programando en 0 y 1

EN los albores de la computación, cuando las computadoras ocupaban habitaciones enteras y los programadores eran considerados una especie de magos modernos, surgió un lenguaje que cambiaría la forma en que nos comunicamos con las máquinas: el lenguaje ensamblador.



Pero, ¿cómo nació esta maravilla de la tecnología? ¿Fue un rayo de inspiración divina o un simple acto de desesperación? Prepárense para un viaje al pasado, donde los bits y los bytes eran los protagonistas y los programadores luchaban por domar a las bestias digitales.

En aquellos tiempos oscuros, los programadores tenían que comunicarse con las computadoras en su idioma nativo: el código binario.

Imagínense tener que escribir un ensayo completo usando solo ceros y unos. No solo era tedioso, sino que también era propenso a errores.

No solo era tedioso, sino que también era propenso a errores.

Un simple error tipográfico podía hacer que la computadora se volviera loca, y créanme, nadie quería ver a una computadora fuera de control en aquellos días. ensamblador.



La llegada de los héroes: Von Neumann, Wilkes y Booth

PERO como en toda buena historia, aparecieron los héroes para salvar el día. John von Neumann, un genio matemático con un cerebro del tamaño de un planeta, sentó las bases de la arquitectura de computadoras modernas.



Su idea de almacenar tanto datos como programas en la misma memoria fue revolucionaria, aunque en ese momento probablemente nadie entendió realmente lo que significaba.

En 1945, este genio húngaro-americano tenía una visión: ¿por qué no crear una

máquina donde las instrucciones y los datos se almacenaran en el mismo lugar? La idea, que luego se llamó la **arquitectura de von Neumann**, consistía en tener una memoria única donde los datos y las instrucciones convivieran juntos.

Aunque von Neumann dio a luz a esta hermosa idea de la computación moderna, también condenó a los programadores a lidiar con instrucciones binarias.

Básicamente, eran como sacerdotes intentando descifrar un código sagrado sin perder la cordura en el proceso.

Pero no desespere, que los buenos inventos siempre nacen de la frustración.

En 1949, un tipo brillante llamado **Maurice Wilkes** estaba trabajando en la EDSAC, una de las primeras computadoras electrónicas de almacenamiento de programas.



Esta máquina era un monstruo de cables y tubos de vacío, pero Wilkes sabía que no iba a poder programarla de manera eficiente usando solo lenguaje de máquina. Después de un par de noches sin dormir y posiblemente varias tazas de té (somos británicos, después de todo), se le ocurrió algo radical: ¿y si creamos un lenguaje que utilice palabras clave en lugar de esos malditos ceros y unos?

Wilkes creó lo que hoy conocemos como uno de los primeros lenguajes ensambladores.

