

## **Historia de la computación**

*Samuel Ricardo López Rosales*

En 1920 Kurt Gödel publico un famoso articulo en el que demuestra que en toda formulación incuestionable: siempre habrá en ella afirmaciones que no se pueden demostrar, pasados los años exactamente en el año 1937 el matemático Alan Turing publico otro articulo en el que desarrollo el teorema que Gödel previamente publicó y es aquí donde se puede considerar el origen oficial de la informática teórica, Alan Turing en este articulo introdujo la maquina de Turing que consta de una cinta tan larga como necesitemos dividida en casillas esta cinta será la memoria y en dicha cinta podemos escribir símbolos por ejemplo 0 y 1, el segundo componente consta de una cabeza capaz de moverse por la cinta hacia la izquierda y hacia la derecha y leer y escribir símbolos en la cinta y finalmente el tercer componente es un programa que le dice a la cabeza que tiene que hacer con cada símbolo. [1]

Después unos años más adelante específicamente en el año 1941 nace el cálculo Lambda este nace a partir de la idea o motivación que tuvo Alonzo Church por saber cómo se podía diseñar un modelo formal para la modelación matemática ósea desde el punto ósea desde el punto de vista computacional cual será esa función y lo que el propone grandes rasgos es que el cálculo Lambda se podría modelar como una caja la cual representa una función, no puedes ver dentro de esa caja ni lo que ocurre en el interior únicamente se pueden introducir valores dentro de esta los cuales serán procesados mágicamente y producen una salida de acuerdo a la entrada. [2]

Otro personaje importante en la historia de la computación fue el matemático llamado John Newman por la creación de una arquitectura en 1945 que dicha arquitectura hoy en día los sistemas informáticos siguen al pie de la letra, esta arquitectura esta formada por: un CPU que no es mas que nada el cerebro del computador, es quien manda y por el pasan todas las instrucciones a cumplirse, otro componente es una memoria principal dividida en dos partes, la primera es la unidad aritmética lógica que es la encargada de almacenar la información

temporalmente que se utilizaran posteriormente y la unidad que administra las entradas y salidas e interacción con el computador. Antes que se desarrollara la arquitectura antes mencionada, John Mackly un físico de la universidad de Transilvania, se propuso construir un ordenador gigante que podría calcular tablas de números en minutos su coste fue de medio millón de dólares, pero aun así se empezó a construir ENIAC que fue denominado un integrador numérico electrónico y ordenador, ENIAC tenía 100 pies de largo y pesaba 30 toneladas, 10,000 resistencias, después de años de trabajo el equipo logro terminar ENIAC este ordenador podía realizar 5000 sumas, 337 multiplicaciones y 38 divisiones cada segundo. [3]

Pasando ya a los 50's comienza en 1955 el desarrollo de FORTRAN, el lenguaje de programación fue liberado en Abril 1957, este lenguaje de programación fue desarrollado principalmente por John Backus en IBM Backus y recibió el premio "Turing Award" en 1977, Backus fue muy criticado porque decían que el desarrollo de FORTRAN solo era un desperdicio de dinero, pero resulto lo contrario fue todo un éxito debido a su excelente documentación y las sofisticadas técnicas de optimización que uso en el diseño de su compilador, y esto porque los programas en FORTRAN se dividen en dos partes: en la parte uno se conocen los valores iniciales y el tipo del valor y en la parte dos las instrucciones a ser ejecutadas [4]. Después en los año 60 se comenzó un proyecto de un lenguaje de programación el cual funciona en todo tipo de maquina y en toda plataforma, este proyecto de denomino COBOL y se usó con el fin de manejar toda gestión de negocio posible. [5]

También el los 60's se empleo el termino maquinas de estado finito, que es un programa de computadora y representa una secuencia de instrucciones a ser ejecutadas, donde cada instrucción depende del estado actual de la maquina y del actual estimulo, de aquí surge el concepto autómeta finito no determinista a manos de los informáticos teóricos Michael Rabin y Dana Scott, este concepto significa que desde un estado tenemos 2 posibles transiciones, para un símbolo de entrada se puede responder de dos maneras diferentes, el autómeta lo que hace es

considerar las dos transiciones y analizar en el resto de los estados y considera por cual camino hay que ir. Después Noam Chomsky publica un trabajo en el que describe las propiedades de los tipos de lenguajes formales y correspondientes gramáticas en relación a su complejidad computacional, Chomsky dividió los lenguajes en tres grupos: regulares, libres de contexto y sensibles al contexto, lo que él pretendía era demostrar que las 2 primeras clasificaciones son difíciles o incapaces de dar cuenta de manera simple de la complejidad de los lenguajes naturales [6].

En el año 1965 Edmonds definió un algoritmo que busca el camino mas corto a un objetivo o el de menor coste, esto condujo a una pregunta fundamental y todavía no resulta y esta es si  $P = PN$ , denominando a  $NP$  = conjunto de problemas que podemos comprobar en un tiempo razonable y  $P$  = conjunto de problemas a los que podemos encontrar la respuesta, y la pregunta surge en que el que podamos comprobar fácilmente una solución no implica que podamos encontrar fácilmente la solución.

En los 70's se da la semántica de los lenguajes de programación, nacen los métodos formales para el estudio de los programas y lenguajes que es lo que hoy en día utilizamos para el desarrollo de software por fases para llegar al objetivo final. También en los años 70 el Dr. Edgar Frank Codd publico un artículo donde propuso que los sistemas que guardan la información deberían estar organizados en estructuras llamadas relaciones, y en 1974 se presento el Ingres el primer manejador de base de datos relacional. Otro avance en esa época fue que surgió Latex, que es un sistema para la maquetación de texto, y es muy utilizado hoy en día para en la producción de documentos científicos y técnicos.

En los 80's surge el modelo de algoritmo paralelo, que es un algoritmo que en vez de ser ejecutado secuencialmente puede ser ejecutado por partes en el mismo instante de tiempo por varias unidades de computo u ordenadores, para finalmente unir todas las partes y obtener un resultado correcto, también surgió un modelo de arquitectura Paralela que son varias computadoras que están interconectadas por

medio de un mecanismo de interconexión, a la par surge un modelo para evaluar que tan complejo es un algoritmo de acuerdo a sus entradas y salidas. [7]

En los 90's se empieza utilizar la criptografía para el intercambio de información, con el fin de que en el traslado no sea modificada o utilizada en otros fines, esto se hace utilizando algoritmos matemáticos para transformar la información en caracteres ilegibles y al llegar a su destino se hace la operación inversa, con esto también se introdujo lo que es la llave pública y la llave privada, que es muy similar cuando vas al SAT, ellos tienen tu llave pública y te dan una llave privada que solo tu utilizarlas. También surge la autenticación que es el acto o proceso de confirmar que algo o alguien es quien dice ser. Y ya pasando a una década más actual llegamos a el año 2000 donde a partir de aquí surgieron grandes cosas que son las que vivimos actualmente por ejemplo surgen los métodos de aproximación que son técnicas cuyo resultado son aproximaciones, se tiene un valor inicial propuesto y de acuerdo a este se va acercando al valor buscado, también surge la minería de datos que es la búsqueda de patrones de acuerdo a datos establecidos para poder predecir eventos y el Big Data que es el manejo de volúmenes grandes de información.

## Bibliografía

- [1] M. Alfonseca, *La máquina de Turing*, Madrid, 2014.
- [2] C. Chacón Sartorí, *Introducción al cálculo Lambda usando Racket*, Valparaíso, 2019.
- [3] M. Barceló, *ARQUITECTURA ¿VON NEUMANN?*, Madrid, 1998.
- [4] A. E. Cáceres Gonzáles, *Lenguajes de programación*, Ciudad de México, 2004.
- [5] R. Lázcares Valenciano y Á. Molina García, *Lenguajes de programación COBOL y PL/I, historia y características principales*, Costa Rica.
- [6] M. Alavi, S. Aliaga y M. Murga, «Máquinas de estado finito,» *Illuminate*, vol. 8, nº 41, pp. 41-47, 2016.
- [7] Departamento de informática, Universidad Valladolid, Campus de Segovia, *Complejidad Algorítmica*, Valladolid.