



TAREA 1.1

Historia y evolución de La Teoría de Autómatas y Lenguajes Formales

INSTITUTO TECNOLÓGICO DE PACHUCA

INGENIERÍA EN SISTEMAS COMPUTACIONALES

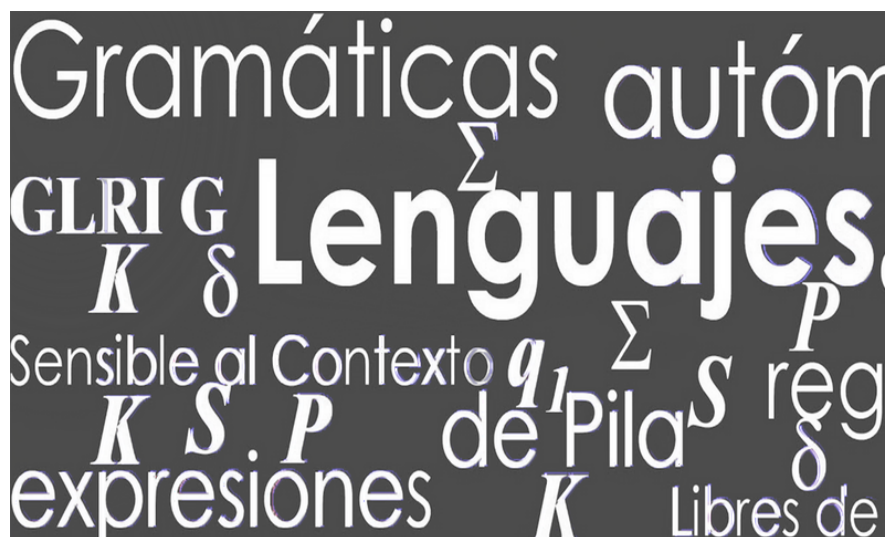
MATERIA:

Lenguajes Autómatas

NOMBRE:

Emanuel Tolentino Santander

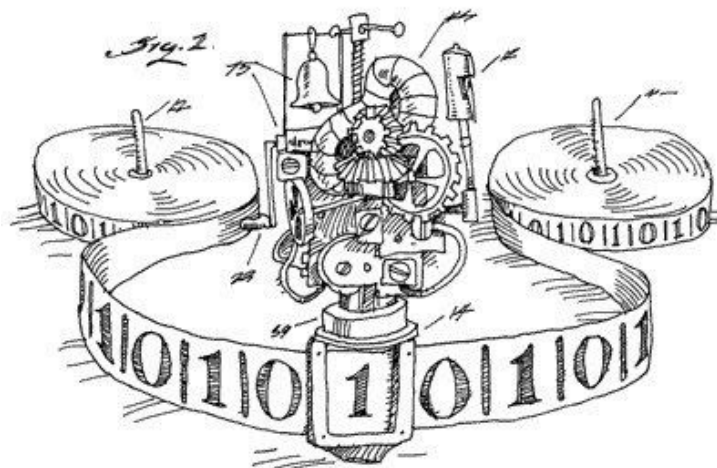
21200635



INTRODUCCION

La Teoría de Autómatas es un campo fundamental en la informática y las matemáticas que se ocupa del estudio abstracto de los sistemas de cómputo. Se centra en comprender y modelar el comportamiento de las máquinas que procesan información siguiendo reglas predefinidas.

La Teoría de Autómatas tiene una amplia gama de aplicaciones en informática y ciencias de la computación, incluyendo el diseño de compiladores, La verificación formal de sistemas, la inteligencia artificial, la bioinformática y la teoría de la complejidad computacional.



INVESTIGADORES QUE SUSTENTARON LA TEORÍA DE AUTÓMATAS

**Año de
Contribución**

Aportación

Alan Turing

1936

Propuso la Máquina de Turing en su artículo seminal de 1936, "On Computable Numbers, with an Application to the Entscheidungsproblem". Esta máquina es un modelo teórico de computación que formaliza el concepto de algoritmo. Fundamentó la noción de computabilidad al demostrar que hay problemas que son indecidibles, es decir, no pueden ser resueltos por un algoritmo.

**Alonzo
Church**

1930

Desarrolló el cálculo lambda como un formalismo para la definición de funciones recursivas y la computabilidad. Demostró que el cálculo lambda y la Máquina de Turing son equivalentes en términos de capacidad computacional, estableciendo así la tesis de Church-Turing.

**Stephen Cole
Kleene**

1950

Introdujo los conceptos de autómatas finitos y expresiones regulares como herramientas para describir lenguajes formales. Estableció la equivalencia entre los autómatas finitos y las expresiones regulares, lo que facilitó el diseño y análisis de lenguajes formales.

INVESTIGADORES QUE SUSTENTARON LA TEORÍA DE AUTÓMATAS

**Año de
Contribución**

Aportación

**John von
Neumann**

1940

Propuso la arquitectura de von Neumann, que separa la memoria y la unidad de procesamiento de una computadora, permitiendo la ejecución de programas almacenados en la memoria. Su trabajo sentó las bases para el diseño de computadoras modernas y la programación de sistemas.

**Noam
Chomsky**

1950

Propuso la jerarquía de Chomsky, que clasifica los lenguajes formales en cuatro tipos (tipos 0, 1, 2 y 3) según su complejidad y el tipo de gramática que los genera. Su trabajo proporcionó una base sólida para entender la complejidad de los lenguajes formales y su relación con los autómatas.

INVESTIGADORES QUE SUSTENTARON LA TEORÍA DE AUTÓMATAS

**Año de
Contribución**

Aportación

Michael Rabin

1960

Contribuyó al desarrollo de la teoría de autómatas no deterministas y la complejidad computacional. Junto con Dana Scott, desarrolló el algoritmo de Rabin-Scott para la minimización de autómatas finitos, un problema fundamental en el diseño de compiladores y la teoría de la computación.

Dana Scott

1960

Pionero en la lógica temporal y la verificación formal de sistemas. Junto con Rabin, desarrolló la lógica temporal lineal y la lógica temporal de Ramificación, que se utilizan en la especificación y verificación de sistemas concurrentes y distribuidos.

CONCLUSION

La Teoría de Autómatas es un campo fascinante que ha desempeñado un papel fundamental en la informática y las matemáticas. A través del estudio de entidades abstractas conocidas como autómatas, este campo nos permite comprender cómo funcionan los sistemas de cómputo y cómo procesan la información siguiendo reglas predefinidas.

Desde los primeros trabajos pioneros de Alan Turing y Alonzo Church hasta las contribuciones más recientes en lógica temporal y verificación formal de sistemas, la Teoría de Autómatas ha evolucionado y se ha diversificado enormemente. Los conceptos y herramientas desarrollados en este campo son ampliamente aplicables, desde el diseño de compiladores hasta la inteligencia artificial y la biología computacional.