

Consulta: Lenguajes Decidibles:

En el contexto de la teoría de la computación, un lenguaje decidable, también conocido como recursivamente enumerable o semi-decidible, se refiere a un conjunto de cadenas (o palabras) sobre un alfabeto determinado para el cual existe un algoritmo que puede decidir si una cadena dada pertenece o no al conjunto.

La definición formal:

Lenguaje Decidable (o recursivamente enumerable): Un lenguaje L es decidable si existe una máquina de Turing que, cuando se le da una cadena de entrada w , siempre se detiene y responde "sí" si w está en L , y "no" de lo contrario. En otras palabras, la máquina de Turing siempre acepta o rechaza una cadena de entrada.

Lenguaje Indecidable: Por otro lado, un lenguaje es indecidible si no existe ninguna máquina de Turing que pueda decidir si una cadena dada pertenece al lenguaje o no. En otras palabras, no hay algoritmo que pueda resolver el problema de pertenencia para todas las cadenas de entrada.

Relacionando esto con máquinas de Turing y autómatas finitos:

Las máquinas de Turing son modelos abstractos de computación que pueden simular el comportamiento de algoritmos en una forma precisa y exhaustiva. Se utilizan para definir formalmente conceptos como decidibilidad e indecidibilidad.

Los autómatas finitos, en particular los autómatas finitos deterministas (AFD) y no deterministas (AFND), son modelos computacionales más simples que las máquinas de Turing, utilizados para reconocer lenguajes regulares. Sin embargo, no son tan poderosos como las máquinas de Turing y no pueden reconocer todos los lenguajes decidibles.

Un lenguaje decidable es aquel para el cual existe un algoritmo (o una máquina de Turing) que puede decidir si una cadena pertenece al lenguaje o no, mientras que un lenguaje indecidible no tiene tal algoritmo. Las máquinas de Turing y los autómatas finitos son herramientas clave para estudiar y comprender la decidibilidad y otros conceptos fundamentales en la teoría de la computación.

Ejemplo:

Lenguaje Decidible:

Consideremos el lenguaje L_{par} , que consiste en todas las cadenas binarias que representan números pares en notación binaria. Es decir, $L_{\text{par}} = \{w \mid w \text{ representa un número par en binario}\}$.

Este lenguaje es decidible porque podemos construir un algoritmo que determine si una cadena dada representa un número par en binario. Por ejemplo, podríamos escribir un programa que, al recibir una cadena binaria como entrada, verifique si el último bit es 0. Si lo es, entonces el número es par; de lo contrario, es impar.

Por lo tanto, el lenguaje L_{par} es decidible.

Lenguaje Indecidible:

Un ejemplo clásico de un lenguaje indecidible es el problema de la detención de la máquina de Turing. Este problema consiste en determinar, dada una descripción de una máquina de Turing y una entrada, si la máquina eventualmente se detendrá o entrará en un bucle infinito cuando se ejecute con esa entrada.

Formalmente, el lenguaje L_{det} es el conjunto de todas las codificaciones de pares ordenados $\langle M, w \rangle$, donde M es una descripción de una máquina de Turing y w es una cadena de entrada, y M se detiene cuando se ejecuta con la entrada w .

El problema de detención es indecidible, lo que significa que no hay un algoritmo que pueda determinar si una máquina de Turing detendrá su ejecución con una entrada dada o entrará en un bucle infinito.

Por lo tanto, el lenguaje L_{det} es indecidible.