



UNIVERSIDAD AUTÓNOMA DE  
CHIHUAHUA

# Problemas no resolubles por una máquina de Turing y problema de la parada

Erick Fernando Nevarez Ávila / 357664

Adrian A. González Domínguez / 359834

Héctor Daniel Medrano Meza / 361345

11 de noviembre de 2024



# ¿Qué son los Problemas No Resolubles?

Problemas resolubles, aquellos los cual no existe algoritmo alguno para resolverlos en tiempo polinómico y No polinómico.



# **Ejemplos de problemas No resolubles**

# Problemas de Correspondencia de Post:

Problema donde se toman 2 pares o mas de palabras y se busca que concatenando las palabras del mismo índice en ambos conjuntos de 1 par de palabras iguales

$u_1$	$u_2$	$u_3$	$u_4$	$v_1$
$aba$	$bbb$	$aab$	$bb$	$a$

$u_1 u_4 u_3 u_1$

$ababbaababa$

$v_1$	$v_2$	$v_3$	$v_4$
$a$	$aaa$	$abab$	$babba$

$v_1 v_4 v_3 v_1$

# Fórmulas Diofantinas (Teorema de Matiyasevich)

Dicho problema busca definir si hay soluciones enteras para un Polinomio

$$x^2 + y^2 = z^2$$



$$x = 3, y = 4, z = 5$$

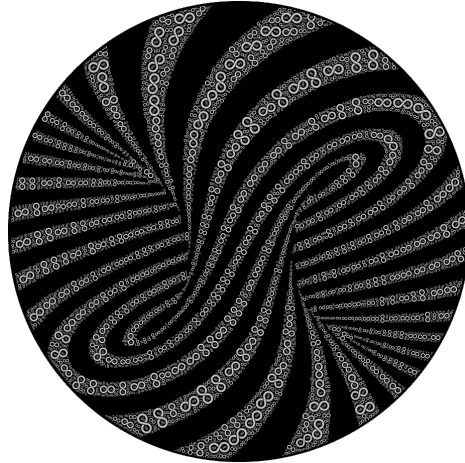
$$-x^3 + 2x^2 + x - 2 = 0$$

¿?

# **El Problema de la Parada**

# Definición

El problema de la parada afirma que no existe un algoritmo capaz de identificar si un programa se detendrá o continuará ejecutándose indefinidamente.



```
while (true) {  
    mostrarMensaje()  
}
```

Algo como esta función no puede determinarse

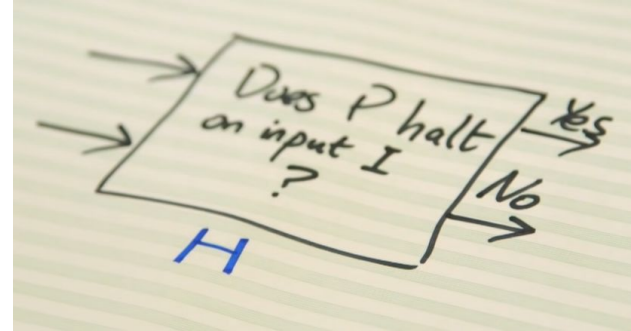
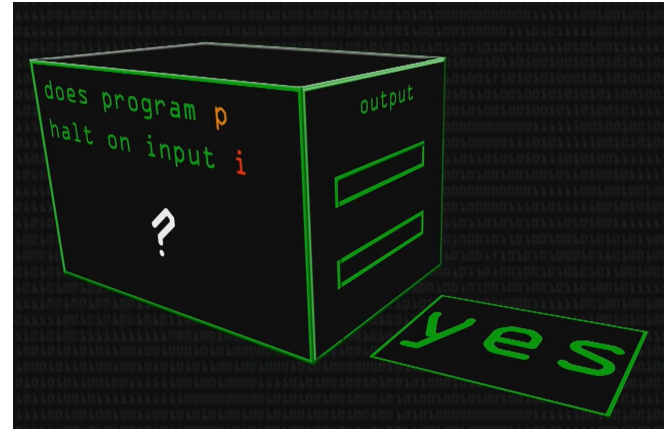
# Fundamento Teórico

Si se asume que el programa existe, tendríamos una máquina, llamada  $H$ , que pide un *algoritmo* y *datos de entrada*.

La pregunta entonces es “*dada una máquina  $H$  con parámetros  $p$ , ¿se detendrá?*”

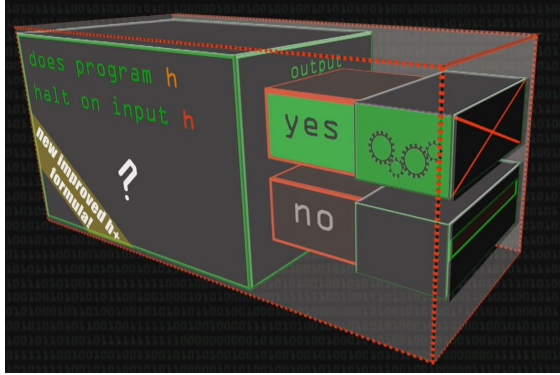
Si la respuesta es **sí**, se cicla eternamente.

Si la respuesta es **no**, se detiene.





# Fundamento Teórico



Sin embargo, si los argumentos que se le pasasen a  $H$  fuese el mismo programa  $H$ , la pregunta se vuelve “*al dársele la entrada  $H$  a la máquina  $H$ , ¿se detendrá?*”

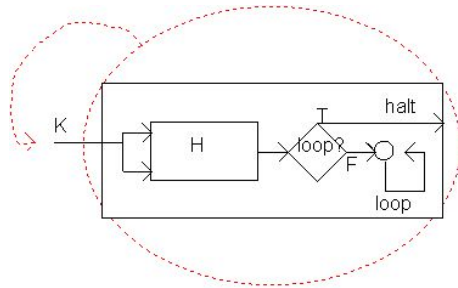
Si la respuesta es que **sí** se detendrá, entonces se cicla. Pero si la respuesta es **no**, entonces se detiene.

```
function Halts(program, input):  
    # Hypothetical function that returns True if 'program' halts on 'input',  
    # and False if it runs indefinitely.  
    ...  
  
function Paradox():  
    if Halts(Paradox, null) == True:  
        # If Halts predicts Paradox will halt, enter an infinite loop.  
        while True:  
            continue  
    else:  
        # If Halts predicts Paradox will not halt, halt immediately.  
        return "Halts"  
  
# Run the paradoxical program  
Paradox()
```

# Fundamento Teórico

Asumimos inicialmente que sí podíamos resolver el problema y es por esto mismo que terminamos en una paradoja, porque nuestra asunción fue incorrecta.

No existe ninguna máquina capaz de resolver el problema de la parada.



# **Impacto en las Ciencias de la Computación**

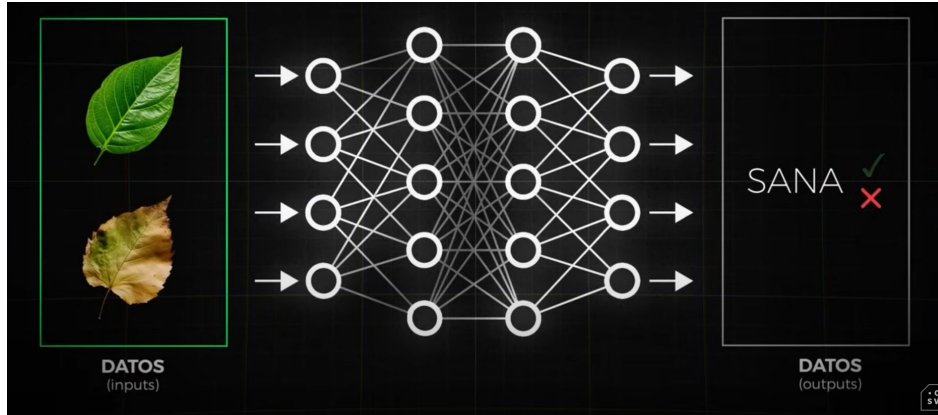
# Limitaciones a la resolución de problemas

No existen algoritmos exactos para todos los problemas del mundo.

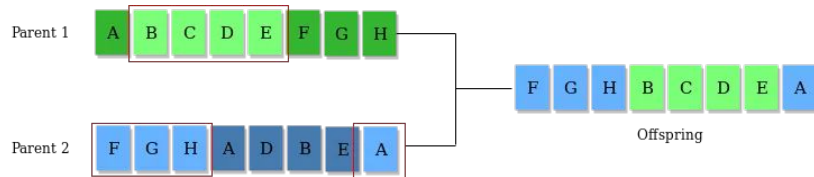
A veces solo podemos aproximar un buen resultado.



# Avances tecnológicos



Teóricamente los problemas son irresolubles, pero esto ha conllevado a la creación de estrategias aproximadas como la IA o los algoritmos genéticos.





# Apego a la realidad



¿Qué planeta tienes en mente ahora mismo?  
La naturaleza no siempre tiene una respuesta a todo.

Muchas veces, una mala respuesta es mejor que una nula respuesta.

# Referencias bibliográficas

Contribuidores de Wikipedia. (2024a, enero 24). *Recursively enumerable language*. Wikipedia.

[https://en.wikipedia.org/wiki/Recursively\\_enumerable\\_language](https://en.wikipedia.org/wiki/Recursively_enumerable_language)

Casero, A. (2024, 15 marzo). ¿Qué son los problemas algorítmicos no resolubles? KeepCoding Bootcamps.

<https://keepcoding.io/blog/problemas-algoritmicos-no-resolubles/>

What unsolved problems in theoretical computer science other than  $P \neq NP$  would make a real world impact in the field? (s. f.). Quora.

<https://www.quora.com/What-unsolved-problems-in-theoretical-computer-science-other-than-P-NP-would-make-a-real-world-impact-in-the-field>

# Referencias bibliográficas

Computerphile. (2014, August 21). Turing & the Halting Problem - Computerphile [Video]. YouTube.

[https://www.youtube.com/watch?v=macM\\_MtS\\_w4](https://www.youtube.com/watch?v=macM_MtS_w4)

Arte de programar. (2022, February 19).  El algoritmo IMPOSIBLE: ¿Qué es el problema Halting? [Video]. YouTube.

<https://www.youtube.com/watch?v=b5bfoC3Z7gs>

Colaboradores de Wikipedia. (2024, August 13). Problema de la parada. Wikipedia, La Enciclopedia Libre.

[https://es.wikipedia.org/wiki/Problema\\_de\\_la\\_parada](https://es.wikipedia.org/wiki/Problema_de_la_parada)