

P vs NP

Un problema muy Complejo

Kevin Cárdenas Gallego



Universidad de Antioquia-FCEN
INSTITUTO DE MATEMÁTICAS

15 de noviembre de 2023



UNIVERSIDAD DE ANTIOQUIA
1803

Tabla de contenido



1 Introducción

- ¿Qué es P vs NP?

2 Preliminares

3 Clase P

4 P vs NP

5 Conclusión



UNIVERSIDAD DE ANTIOQUIA
1803

- El problema P vs NP es uno de los siete problemas del milenio propuestos por el Clay Mathematics Institute en el año 2000.



- El problema P vs NP es uno de los siete problemas del milenio propuestos por el Clay Mathematics Institute en el año 2000.
- El problema P vs NP es uno de los problemas más importantes en la ciencia de la computación.



Tabla de contenido



- 1 Introducción
 - ¿Qué es P vs NP?
- 2 Preliminares
- 3 Clase P
- 4 P vs NP
- 5 Conclusión



UNIVERSIDAD DE ANTIOQUIA
1803

¿Qué es P vs NP?



- El problema P vs NP se pregunta si los problemas que se pueden verificar en tiempo polinomial también se pueden resolver en tiempo polinomial.



¿Qué es P vs NP?



- El problema P vs NP se pregunta si los problemas que se pueden verificar en tiempo polinomial también se pueden resolver en tiempo polinomial.
- Verificar la solución de un problema significa que se puede comprobar si una solución es correcta o no.



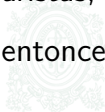
¿Qué es P vs NP?



- El problema P vs NP se pregunta si los problemas que se pueden verificar en tiempo polinomial también se pueden resolver en tiempo polinomial.
- Verificar la solución de un problema significa que se puede comprobar si una solución es correcta o no.
- Resolver un problema significa encontrar una solución correcta.



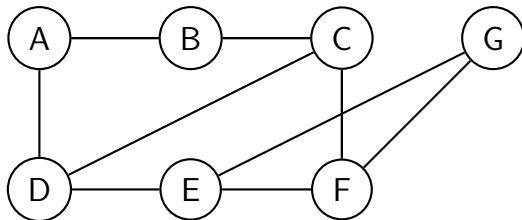
Por ejemplo el problema de encontrar un camino más corto entre dos nodos en un grafo. Si se tiene un camino entre dos nodos, se puede verificar si es el camino más corto, pero no se sabe si se puede encontrar en tiempo polinomial. Para verificarlo se debe recorrer el camino y sumar los pesos de las aristas, si el resultado es menor que el peso del camino que se quiere verificar, entonces se encontró un camino más corto.



Ejemplo



Tomemos un grafo G con los nodos A, B, C, D, E, F y las aristas con pesos homogéneos como se muestra en la siguiente figura:



Si tenemos un camino entre A y G como el de la figura anterior, se puede verificar. Siga el siguiente algoritmo:



Ejemplo



- 1 tome todos los posibles caminos entre A y G .



Ejemplo



- 1 tome todos los posibles caminos entre A y G .
- 2 calcule el peso de cada camino.



Ejemplo



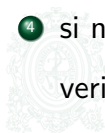
- 1 tome todos los posibles caminos entre A y G .
- 2 calcule el peso de cada camino.
- 3 si el peso de un camino es menor que el peso del camino que se quiere verificar, entonces se encontró un camino más corto.



Ejemplo



- 1 tome todos los posibles caminos entre A y G .
- 2 calcule el peso de cada camino.
- 3 si el peso de un camino es menor que el peso del camino que se quiere verificar, entonces se encontró un camino más corto.
- 4 si no se encontró un camino más corto, entonces el camino que se quiere verificar es el más corto.



Esta certificación, es decir, este algoritmo, tiene un tiempo exponencial, ya que el número de caminos entre dos nodos es exponencial. la pregunta aquí es si se puede encontrar un camino más corto en tiempo polinomial.



Tabla de contenido



- 1 Introducción
 - ¿Qué es P vs NP?
- 2 Preliminares
- 3 Clase P
- 4 P vs NP
- 5 Conclusión



UNIVERSIDAD DE ANTIOQUIA
1803

- Para entender el problema P vs NP, primero se debe entender la clase de problemas P y la clase de problemas NP.



- Para entender el problema P vs NP, primero se debe entender la clase de problemas P y la clase de problemas NP.
- La clase de problemas P es la clase de problemas que se pueden resolver en tiempo polinomial.



- Para entender el problema P vs NP, primero se debe entender la clase de problemas P y la clase de problemas NP.
- La clase de problemas P es la clase de problemas que se pueden resolver en tiempo polinomial.
- La clase de problemas NP es la clase de problemas que se pueden verificar en tiempo polinomial.

Definición (Algoritmo)

Un algoritmo es una secuencia de pasos que se siguen para resolver un problema.



Definición (Algoritmo)

Un algoritmo es una secuencia de pasos que se siguen para resolver un problema.

Definición (Tiempo de ejecución)

El tiempo de ejecución de un algoritmo es el número de pasos que se siguen para resolver un problema. Cuando se habla de costo de un algoritmo, se refiere al tiempo de ejecución.

Observación

El tiempo de ejecución de un algoritmo depende de la entrada. Conocer un algoritmo para hallar soluciones de un problema no significa que sea el más óptimo. Es importante notar verificar una solución es igual o menos costoso que encontrarla.



Definición (Tiempo polinomial)

Un algoritmo tiene tiempo polinomial si el tiempo de ejecución es una función polinomial del tamaño de la entrada.



Definición (Tiempo polinomial)

Un algoritmo tiene tiempo polinomial si el tiempo de ejecución es una función polinomial del tamaño de la entrada.

Definición (Tamaño de la entrada)

El tamaño de la entrada es el número de bits que se necesitan para representar la entrada.



Observación

Vamos a tratar sólo funciones de tuplas de naturales (\mathbb{N}^t) en naturales (\mathbb{N}).



Observación

Vamos a tratar sólo funciones de tuplas de naturales (\mathbb{N}^t) en naturales (\mathbb{N}).

Definición (Problema de decisión)

Un problema de decisión es un problema que tiene como respuesta sí o no.



Ejemplo



Por ejemplo el problema de decidir si un número es primo o no. esta se puede escribir como una función



Ejemplo



Por ejemplo el problema de decidir si un número es primo o no. esta se puede escribir como una función

$$\text{Primo} : \mathbb{N} \longrightarrow \{0, 1\}$$

$$n \longmapsto \begin{cases} 1 & \text{si } n \text{ es primo} \\ 0 & \text{si } n \text{ no es primo} \end{cases}$$



Definición (Instancia de un problema de decisión)

Una instancia de un problema de decisión es una entrada para el problema de decisión.



Definición (Instancia de un problema de decisión)

Una instancia de un problema de decisión es una entrada para el problema de decisión.

Por ejemplo, la instancia del problema de decidir si un número es primo, o no, es un número natural.

Definición (Solución de un problema de decisión)

Una solución de un problema de decisión es la respuesta al problema de decisión.

Tabla de contenido



- 1 Introducción
 - ¿Qué es P vs NP?
- 2 Preliminares
- 3 Clase P
- 4 P vs NP
- 5 Conclusión



UNIVERSIDAD DE ANTIOQUIA
1803

Definición (Clase P)

La clase P es la clase de problemas de decisión que se pueden resolver en tiempo polinomial.



Definición (Clase P)

La clase P es la clase de problemas de decisión que se pueden resolver en tiempo polinomial.

Decidir si un número es primo o no, se puede resolver en tiempo polinomial.



Siga el siguiente algoritmo:



Siga el siguiente algoritmo:

- 1 si $n = 1$ entonces n no es primo.



Siga el siguiente algoritmo:

- 1 si $n = 1$ entonces n no es primo.
- 2 para i entre 2 y $n - 1$:



Siga el siguiente algoritmo:

- ① si $n = 1$ entonces n no es primo.
- ② para i entre 2 y $n - 1$:
 - si i divide a n entonces n no es primo.



Siga el siguiente algoritmo:

- ➊ si $n = 1$ entonces n no es primo.
- ➋ para i entre 2 y $n - 1$:
 - si i divide a n entonces n no es primo.
- ➌ n es primo.

Definición (Clase NP)

La clase NP es la clase de problemas de decisión que se pueden verificar en tiempo no polinomial.



Definición (Clase NP)

La clase NP es la clase de problemas de decisión que se pueden verificar en tiempo no polinomial.

Decidir si un número es primo o no, se puede verificar en tiempo no polinomial.



Siga el siguiente algoritmo:



Siga el siguiente algoritmo:

- 1 si $n = 1$ entonces n no es primo.



Siga el siguiente algoritmo:

- 1 si $n = 1$ entonces n no es primo.
- 2 Haga 2^n cosas.



Siga el siguiente algoritmo:

- 1 si $n = 1$ entonces n no es primo.
- 2 Haga 2^n cosas.
- 3 para i entre 2 y $n - 1$:



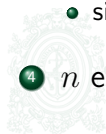
Siga el siguiente algoritmo:

- ❶ si $n = 1$ entonces n no es primo.
- ❷ Haga 2^n cosas.
- ❸ para i entre 2 y $n - 1$:
 - si i divide a n entonces n no es primo.



Siga el siguiente algoritmo:

- ➊ si $n = 1$ entonces n no es primo.
- ➋ Haga 2^n cosas.
- ➌ para i entre 2 y $n - 1$:
 - si i divide a n entonces n no es primo.
- ➍ n es primo.



Definición (Certificado)

Un certificado es una prueba de que una solución es correcta.



Definición (Certificado)

Un certificado es una prueba de que una solución es correcta.

Por ejemplo, si se quiere verificar que $n = 7$ es primo, se puede dar como certificado que 7 es sólo divisible por 1 y 7.



Definición (Certificado)

Un certificado es una prueba de que una solución es correcta.

Por ejemplo, si se quiere verificar que $n = 7$ es primo, se puede dar como certificado que 7 es sólo divisible por 1 y 7.

Observación

El certificado no es la solución, es una prueba de que la solución es correcta.



Tabla de contenido



- 1 Introducción
 - ¿Qué es P vs NP?
- 2 Preliminares
- 3 Clase P
- 4 P vs NP
- 5 Conclusión



UNIVERSIDAD DE ANTIOQUIA
1803

Enunciado del problema



Definición (P vs NP)

El problema P vs NP se pregunta si los problemas que se pueden verificar en tiempo polinomial también se pueden resolver en tiempo polinomial.



Enunciado del problema



Definición (P vs NP)

El problema P vs NP se pregunta si los problemas que se pueden verificar en tiempo polinomial también se pueden resolver en tiempo polinomial.

Observación

El problema P vs NP es un problema de decisión. Tiene mucho que ver con complejidad computacional. Pues se pregunta sobre algoritmos optimos, conteo de operaciones y tiempo de ejecución.

Considere el problema de buscar un grupo de números entre cierto conjunto de números que sumen 0. Si se tiene un grupo de números que sumen 0, se puede verificar en tiempo polinomial, pero encontrar el grupo de números que sumen 0, hasta el momento no se sabe si se puede encontrar en tiempo polinomial.



Tabla de contenido



- 1 Introducción
 - ¿Qué es P vs NP?
- 2 Preliminares
- 3 Clase P
- 4 P vs NP
- 5 Conclusión



UNIVERSIDAD DE ANTIOQUIA
1803

- El problema P vs NP es uno de los problemas más importantes en la ciencia de la computación.



- El problema P vs NP es uno de los problemas más importantes en la ciencia de la computación.
- El problema P vs NP se pregunta si los problemas que se pueden verificar en tiempo polinomial también se pueden resolver en tiempo polinomial.



- El problema P vs NP es uno de los problemas más importantes en la ciencia de la computación.
- El problema P vs NP se pregunta si los problemas que se pueden verificar en tiempo polinomial también se pueden resolver en tiempo polinomial.
- Parece intuitivo pensar que no es cierto, es decir que P es distinto de NP, pero hasta el momento no se ha podido demostrar.

 Wikipedia.

Problema P contra NP,

Consultado el 14 de noviembre de 2023.

 Canal de YouTube.

Título del Video,

Consultado el 14 de noviembre de 2023.





Gilles Brassard and Paul Bratley

Algorithmics: Theory and Practice

Prentice Hall, 1988.



UNIVERSIDAD DE ANTIOQUIA
1803