

# Árbol Parcial mínimo de Prim.

INTELIGENCIA ARTIFICIAL

Jose Cruz TP  
CETI COLOMOS

### *¿Qué es el árbol parcial mínimo de Prim?*

El árbol parcial mínimo de Prim es un algoritmo para encontrar un árbol recubierto mínimo en un grafo conexo, no dirigido y cuyas aristas están etiquetadas. Un árbol recubierto mínimo es un árbol que conecta todos los vértices del grafo y tiene el peso total mínimo posible.

### *¿Para qué sirve el árbol parcial mínimo de Prim?*

El árbol parcial mínimo de Prim se utiliza en una amplia gama de aplicaciones, que incluyen:

Redes: El árbol parcial mínimo de Prim se utiliza para encontrar la ruta más corta entre dos puntos en una red, como una red de carreteras o una red de telecomunicaciones.

Distribución: El árbol parcial mínimo de Prim se utiliza para encontrar la ruta más eficiente para distribuir bienes o servicios.

Construcción: El árbol parcial mínimo de Prim se utiliza para encontrar la ruta más económica para construir una red de infraestructura, como una red de carreteras o una red de tuberías.

### *¿Cómo se implementa el árbol parcial mínimo de Prim en el mundo?*

El árbol parcial mínimo de Prim se implementa en una amplia gama de productos y servicios, que incluyen:

Navegadores GPS: Los navegadores GPS utilizan el árbol parcial mínimo de Prim para encontrar la ruta más corta entre dos puntos.

Sistemas de entrega: Los sistemas de entrega utilizan el árbol parcial mínimo de Prim para encontrar la ruta más eficiente para entregar productos.

Software de diseño de redes: El software de diseño de redes utiliza el árbol parcial mínimo de Prim para encontrar la ruta más económica para construir una red de infraestructura.

### *¿Cómo implementaría el árbol parcial mínimo de Prim en mi vida?*

Podría implementar el árbol parcial mínimo de Prim en mi vida para encontrar la ruta más corta entre dos puntos en mi ciudad. Por ejemplo, podría usar el algoritmo para encontrar la ruta más rápida para ir del trabajo a casa o para encontrar la ruta más corta para visitar a un amigo.

### *¿Cómo implementaría el árbol parcial mínimo de Prim en mi trabajo o en mi trabajo de ensueño?*

En mi trabajo actual, podría utilizar el árbol parcial mínimo de Prim para optimizar las rutas de entrega de los productos. Por ejemplo, podría usar el algoritmo para encontrar la ruta más eficiente para entregar un paquete a un cliente.

En mi trabajo de ensueño, podría utilizar el árbol parcial mínimo de Prim para ayudar a las personas a encontrar el camino más corto para acceder a servicios esenciales, como atención médica o educación. Por ejemplo, podría usar el algoritmo para encontrar la ruta más rápida para llegar a un hospital o para encontrar la ruta más corta para llegar a una escuela.

En general, el árbol parcial mínimo de Prim es una herramienta poderosa que se puede utilizar para encontrar la ruta más corta entre dos puntos en una amplia gama de aplicaciones.

### *Ejemplo de implementación en la vida real*

Imagine que vive en una ciudad con un sistema de transporte público. El sistema de transporte público está formado por una red de autobuses, trenes y tranvías.

Para encontrar la ruta más corta entre dos puntos en la ciudad, podría utilizar el árbol parcial mínimo de Prim para encontrar un árbol recubierto mínimo de la red de transporte público. Este árbol recubierto mínimo representaría la ruta más eficiente para viajar entre cualesquiera dos puntos de la ciudad.

El árbol parcial mínimo de Prim podría implementarse utilizando un algoritmo de programación dinámica. El algoritmo comenzaría con un árbol vacío que consta de un solo vértice, que representa el punto de partida. Luego, el algoritmo agregaría un vértice a la vez al árbol, siempre que el vértice agregado no creara un ciclo en el árbol. El vértice agregado sería el vértice que tiene la menor distancia a un vértice ya en el árbol.

El algoritmo continuaría agregando vértices al árbol hasta que todos los vértices de la red de transporte público estuvieran incluidos en el árbol. El árbol resultante sería el árbol recubierto mínimo de la red de transporte público.

Este árbol recubierto mínimo podría utilizarse para encontrar la ruta más corta entre cualesquiera dos puntos de la ciudad. Por ejemplo, si quisiera encontrar la ruta más corta para ir del trabajo a casa, podría utilizar el árbol recubierto mínimo para encontrar la ruta que pasa por la menor cantidad de autobuses, trenes y tranvías.

Ejemplo:

Spyder (Python 3.11)

File Edit Search Source Run Debug Consoles Projects Tools View Help

C:\Users\Jose Cruz TP\Documents\Python Scripts\untitled1.py

```
23 # Encuentra el nodo con la distancia mínima no visitado.
24 min_key = sys.maxsize
25 min_index = -1
26 for v in range(num_vertices):
27     if not visited[v] and key[v] < min_key:
28         min_key = key[v]
29         min_index = v
30
31 visited[min_index] = True
32
33 # Agregamos el borde a la lista del MST.
34 if mst[min_index] is not None:
35     print(f"Arista: {mst[min_index]} - Peso: {key[min_index]}")
36
37 for v in range(num_vertices):
38     if (
39         graph[min_index][v] != 0
40         and not visited[v]
41         and graph[min_index][v] < key[v]
42     ):
43         key[v] = graph[min_index][v]
44         mst[v] = f"{min_index}-{v}"
45
46 # Ejemplo de grafo ponderado en forma de matriz de adyacencia.
47 graph = [
48     [0, 2, 0, 6, 0],
49     [2, 0, 3, 8, 5],
50     [0, 3, 0, 0, 7],
51     [6, 8, 0, 0, 9],
52     [0, 5, 7, 9, 0],
53 ]
54
55 print("Pasos para construir el Árbol Parcial Mínimo de Prim:")
56 prim_mst(graph)
```

Name	Type	Size	Value
current_node	NoneType	1	NoneType object
distances	dict	4	{'A':0, 'B':1, 'C':3, 'D':4}
end_node	str	1	D
graph	list	5	[[0, 2, 0, 6, 0], [2, 0, 3, 8, 5], [0, 3, 0, 0, 7], [6, 8, 0, 0, 9], [0, 5, 7, 9, 0]]
previous_nodes	dict	3	{'B':'A', 'C':'B', 'D':'C'}
start_node	str	1	A

Help Variable Explorer Plots Files

Console 1/A

Nodo actual: A, Distancia desde A: 0  
Camino más corto desde A a D: ['A', 'B', 'C', 'D']

```
In [2]: runfile('C:/Users/Jose Cruz TP/Documents/Python Scripts/untitled1.py',
wdir='C:/Users/Jose Cruz TP/Documents/Python Scripts')
Pasos para construir el Árbol Parcial Mínimo de Prim:
Arista: 0-1 - Peso: 2
Arista: 1-2 - Peso: 3
Arista: 1-4 - Peso: 5
Arista: 0-3 - Peso: 6

In [3]:
```

Python Console History

conda (Python 3.11.4) ✓ Completions: conda ✓ LSP: Python Line 21, Col 1 UTF-8 CRLF RW Mem 64%

Windows Taskbar: Descargas, General (Visió..., IA\_P2-PRA/Ár..., Spyder (Pytho..., Documento1..., 6° E\_21110353...

System Tray: 11:46 a. m., 26/10/2023