

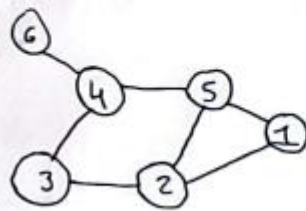
Investigación 04

Grafo, un poco de su historia...

El origen de la teoría de grafos se remonta al siglo XVIII con el problema de los puentes de Königsberg, el cual consistía en encontrar un camino que recorriera los 7 puentes del río Pregel en la ciudad de Königsberg, actualmente Kaliningrado, de modo que recorrieran todos los puentes pasando una sola vez por ellos. El trabajo de Euler en 1736 se considera el primer resultado de la teoría de grafos.

Concepto de grafo

Un grafo en el ámbito de la ciencia de la computación es un tipo de dato abstracto que consiste en un conjunto de nodos (también llamados vértices) y un conjunto de arcos (aristas) que establecen relaciones entre los nodos.



Un grafo de 6 vértices y 7 aristas

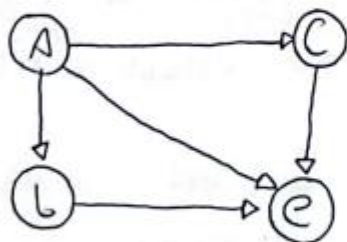
Comunmente estos son utilizados para el modelado de problemas, pues un grafo es un conjunto no vacío de objetos o entes físicos que tienen relación entre ellos.

Un grafo aparece con gran frecuencia como respuesta a problemas de la vida cotidiana: por ejemplo, un gráfico de una serie de tareas a realizar indicando su secuenciación (un organigrama), una red de carreteras o red de enlaces ferroviarios/aéreos.

En pocas palabras, un grafo es un objeto geométrico/combinatorio, es decir, un conjunto de puntos y un conjunto de líneas que une cada par de vertices.

Grafo dirigido

Este es aquel en el que todas sus aristas tienen sentido y dirección.



En este ejemplo

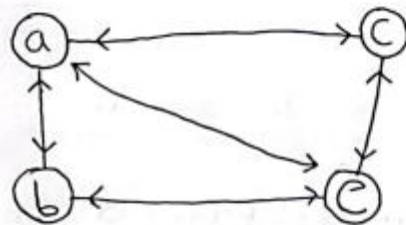
$$V = \{a, b, c, d\}$$

$$E = \{(a, b), (a, c), (a, d), (b, d), (c, d)\}$$

Por otro lado...

Grafo no dirigido

En este, las aristas representan relaciones simétricas y no tienen sentido definido (caso contrario al grafo dirigido en el cual las aristas sí tienen un sentido y no necesariamente son simétricas). Con el término simétrico se hace referencia a que



En este ejemplo

$$V = \{a, b, c, d\}$$

$$E = \{(a, b), (a, c), (a, d), (b, d), (c, d)\}$$

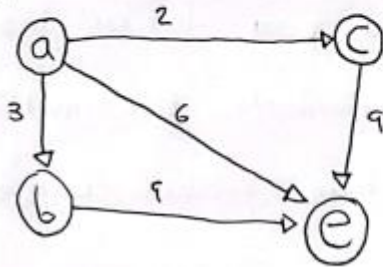
Tanto a las aristas como vertices se les puede asociar información.

A esta información se le llama etiqueta. Si la etiqueta que se asocia es un número se le denomina peso, costo o longitud.

Entonces, un grafo cuyas aristas o vertices tienen pesos asociados recibe el nombre de grafo etiquetado o ponderado.

Grafo ponderado

Un grafo ponderado o con costos es un grafo donde cada arista tiene asociado un valor o etiqueta, para representar el costo, peso, longitud etc.



Operaciones más básicas de los gráficos

Solo como mini resumen, los grafos son estructuras de programación que permiten representar relaciones entre elementos de manera visual y eficiente.

• Construcción del grafo:

Para crearlo, se debe definir previamente la estructura de datos adecuada para representarlo.

Ya sea una matriz de adyacencia o una lista de adyacencia. Ya decidido, solo hace falta inicializar sus punteros, peso y atributos.

- Búsqueda de un nodo:

Para buscar un nodo en un grafo, se pueden recorrer todos los nodos del grafo hasta encontrar al deseado o se pueden usar algoritmos de búsqueda más eficientes como la búsqueda por profundidad o la búsqueda por anchura.

- Adición de un nodo:

Para agregar un nodo a un grafo, se debe crear un nuevo nodo y agregarlo a la estructura de datos que representa el grafo, posteriormente se deben hacer las conexiones correspondientes entre el nuevo nodo y los nodos existentes.

- Eliminación de un nodo

Para eliminar un nodo de un grafo, se deben eliminar todas las conexiones que lo vinculan a otros nodos y luego eliminar la estructura de datos que representa el grafo.

Adición de una conexión

Para agregar, se deben agregar los nodos al nuevo dato/representación de dato y luego agregar la conexión correspondiente.

Eliminación de una conexión

Para eliminar una conexión, se debe buscar la conexión en la estructura de datos que represente el grafo y posteriormente eliminarla.

Encontrar el camino más corto

Existen algoritmos como el de Dijkstra o Bellman-Ford que permiten encontrar la ruta más corta de la manera más eficiente.

Dijkstra - Pseudocódigo

1. Inicializar todas las distancias a infinito y el nodo origen a 0.
2. Crear un conjunto vacío de nodos visitados
3. Crear un conjunto de nodos no visitados que contenga todos los nodos del grafo.
4. Mientras haya nodos no visitados
Encuentra el nodo no visitado con la distancia mínima
Para cada nodo vecino del actual que este en el conjunto no visitado
Calcular la distancia del actual al vecino.

Bibliografía

- Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2009). Introduction to Algorithms (3rd ed.). MIT Press
- Sedgewick, R., & Wayne, K. (2011). Algorithms (4th ed.). Addison-Wesley.
- https://posgrados.inaoep.mx/archivos/PosCsComputacionales/Curso_Propedeutico/Programacion_Estructuras_Datos/Capitulo_10_Grafos.pdf
- Ciencias computacionales Propedéutico: Programacion. (s. f.). INADE. Recuperado 12 de marzo de 2023, de https://posgrados.inaoep.mx/archivos/PosCsComputacionales/Curso_Propedeutico/Programacion_Estructuras_Datos/Capitulo_10_Grafos.pdf
- GRAFOS. (s. f.). <https://ccia.ugr.es/%7Ejfv/ed1/c++/cdrom4/paginaWeb/grafos.htm>
- DSTool: Herramienta para la programación con estructuras de datos. (s. f.). <http://www.hci.uniovi.es/Products/DSTool/grafos/grafos-queSon.html>
- INSTITUTO TECNOLÓGICO DE NUEVO LAREDO DEPTO. DE SISTEMAS Y COMPUTACIÓN. (s. f.). <http://fcaenlinea1.unam.mx/>. Recuperado 12 de marzo de 2023, de http://fcaenlinea1.unam.mx/anexos/1566/1566_u4_anexo3.pdf