

### **Actividad 4.3 – Actividad Integral de Grafos**

Mauricio Cantú Torres A01633805

Joel Isaías Solano Ocampo A01639289

Programación de estructuras de datos y algoritmos fundamentales

Grupo 12

Instituto Tecnológico y de Estudios Superiores de Monterrey

Domingo 28 de noviembre de 2021

### **Actividad 4.3 – Actividad Integral de Grafos**

Para esta actividad, se solicita la elaboración de un programa que lea un archivo de entrada y almacene los datos en una lista de adyacencias que esté organizada por la dirección de IP origen. Después, el programa debe determinar el grado de cada nodo del grafo para después realizar una comparación entre ellos, esto con el propósito de obtener las 5 IP's con los grados más altos.

#### **Grafos**

Los grafos son una estructura de datos abstractos que suelen utilizarse para representar relaciones complejas y no lineales entre diferentes objetos. Un grafo está compuesto por nodos y están conectados por arcos. Existen 3 tipos de grafos principales, los dirigidos, los no dirigidos y los etiquetados. Primeramente, los grafos no dirigidos poseen la capacidad de permitir el movimiento en cualquier dirección de los arcos, mientras que un grafo dirigido tiene arcos que tienen una sola dirección establecida. También, existen los grafos etiquetados, la diferencia entre este tipo de grafo y los anteriores es que los grafos etiquetados tienen un valor asociado a sus arcos, a este valor se le llama “peso”. El peso tiene la función de registrar información respecto a los arcos de un grafo y tiene diferentes aplicaciones en un problema real.

#### **Complejidad temporal**

Tomando en cuenta que para esta actividad se optó por utilizar una lista de adyacencia, la complejidad temporal de la estructura de datos cambia dependiendo de la interacción que se tenga con la lista. Por ejemplo, para añadir un nodo, la complejidad temporal sería constante, al igual que cuando se desea añadir un arco. Por otro lado, cuando se desea remover un nodo, la complejidad temporal es de  $O(V + E)$ , y cuando se quita un arco, la complejidad temporal es de  $O(E)$ .

#### **Referencias**

Graph. (s.f.). Isaac Computer Science. Recuperado de [https://isaaccomputerscience.org/concepts/dsa\\_datastruct\\_graph?examBoard=all&stage=all](https://isaaccomputerscience.org/concepts/dsa_datastruct_graph?examBoard=all&stage=all)