

---

## Estructura de Datos y Algoritmos Fundamentales

Tecnológico de Monterrey Campus Guadalajara

Emilio González Martínez A01640120

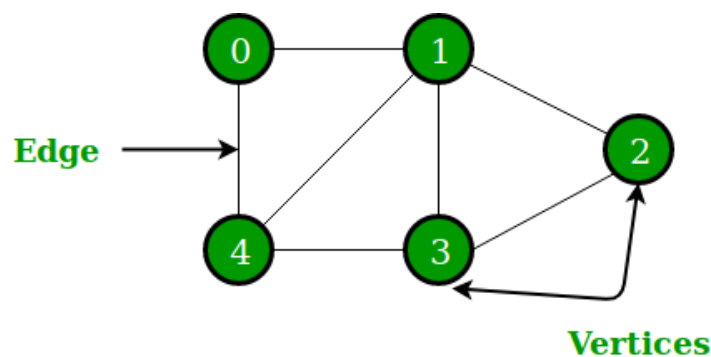
Gael Rodríguez Gómez A01639279

### Act 4.3 - Actividad Integral de Grafos (Evidencia Competencia)

22 de noviembre de 2021

#### Grafos

Los grafos son una composición de conjuntos de nodos. En ellos se almacenan diferentes tipos de elementos o datos que podemos utilizar para procesar o conocer con fines específicos. Adicionalmente estos nodos, suelen estar unidos o conectados a otros nodos a través de elementos que denominamos aristas.



#### Tipos de grafos

- **Grafo no dirigido:** No tienen ninguna dirección particular, son bidireccionales.
- **Grafo Dirigido:** Sus arcos tienen una dirección en particular, con un origen y destino.
- **Grafo Ponderado:** Los arcos en el grafo tienen un valor al cuál denominamos peso.

#### Ventajas de los grafos

Las razones por las que los grafos son ampliamente utilizados es porque son muy flexibles y se pueden adaptar a cualquier situación, pueden modelar diversos fenómenos y acomodos de datos. Además, las operaciones que se pueden realizar en esta estructura permiten resolver problemas de encontrar el camino más corto, de trazado de trayectorias, de optimización más

---

que nada, lo que los hace muy valiosos al momento de desarrollar proyectos, donde generar soluciones óptimas que le permita a los sistemas tomar mejores decisiones es esencial. Otro aspecto muy positivo que tienen es que la mayoría de sus operaciones no llegan ni siquiera a la complejidad  $O(n \log n)$ , por lo que se consideran una estructura eficiente.

## Aplicación

Los gráficos se utilizan para resolver muchos problemas de la vida real. Por ejemplo, los gráficos se utilizan para representar redes. Las redes pueden incluir trayectos en una ciudad, una red telefónica o una red de circuitos. Cada nodo es una estructura y contiene información acerca de la red que se visita.

En nuestra actividad integradora, se nos presenta una serie de registros que contiene diferentes IP que se ven involucradas en ataques cibernéticos, por lo que nuestra tarea es realizar un análisis de estas redes para lograr descubrir la red con mayor ataque, la cual denominamos, **boot master**, para esto, nos servirá implementar una estructura de Grafos, y poder realizar un registro de nuestras IP, y las IP que están atacando, así, después de realizar un segundo registro de estas, por medio de la estructura de datos MaxHeap realizar el acomodo y obtener las cinco IP que más ataques tienen, de las cuales, todas las que tengan la misma cantidad que el máximo número de ataques en registro, se denominarán posibles boot master.

El tipo de grafo que utilizamos en esta actividad es un grafo dirigido no ponderado de representación por lista de adyacencias, ya que los ataques entre las redes no tienen ningún peso y la relación que se hace entre las mismas es únicamente de una red perpetradora a una red victimizada, la lista de adyacencia nos permite ahorrar mucha memoria ya que nos evita crear una matriz cuadrada de tamaño igual al de redes que se tengan y la probabilidad de que la mayoría de estas se conecten entre sí no es suficiente como para decantarnos por esta.

Consideramos que esta estructura de datos es muy eficiente y apropiada para una situación de esta naturaleza, ya que nos permite saber cuáles IP están atacando y a cuáles, a través de esto podemos visualizar y percibir cuales son las redes más infectadas o que están siendo interferidas más seguido, esencial al momento de tomar decisiones respecto a cómo atacar una problemática tan seria y en la que se tienen miles de redes, y por lo tanto, de posibles decisiones. A pesar de que esta actividad no resuelve la problemática como tal, otorga la información necesaria para llegar a una resolución de forma eficiente y eficaz.

---

## Eficiencia

Nuestro grafo está implementado por medio de una lista de adyacencia, por lo que en el peor de los casos, sus operaciones serán lineales,  $O(n)$ , lo cuál lo hace muy eficiente.

Una vez se leyeron todas las IP por integrar al grafo y antes de empezar a agregar adyacencias, realizamos un ordenamiento por método heapsort, que tiene complejidad temporal  $O(n \log n)$  y va a permitir buscar redes de forma más eficiente, lo cual resulta muy útil para registrar los ataques de forma rápida y ahorra mucho tiempo de ejecución al momento de querer agregar una adyacencia tras leer un reporte, especialmente cuando se tienen miles y miles de registros por leer.

Para la búsqueda de índices de nuestras IP, tenemos el método binary search, el cuál tiene una eficiencia temporal de  $O(\log n)$  y podemos realizarlo gracias al ordenamiento hecho previamente, lo cual se combina con nuestro método para agregar adyacencias el cual es de complejidad  $O(1)$ , lo que hace que, en general, la eficiencia para agregar reportes sea muy buena, gracias al ordenamiento hecho previamente.

## Referencias

O. (2021a, febrero 11). Grafos | Qué son, tipos, orden y herramientas de visualización. GraphEverywhere.

<https://www.grapheverywhere.com/grafos-que-son-tipos-orden-y-herramientas-de-visualizacion/>

GeeksforGeeks. (2021b). Graph Data Structure And Algorithms.

<https://www.geeksforgeeks.org/graph-data-structure-and-algorithms/>