

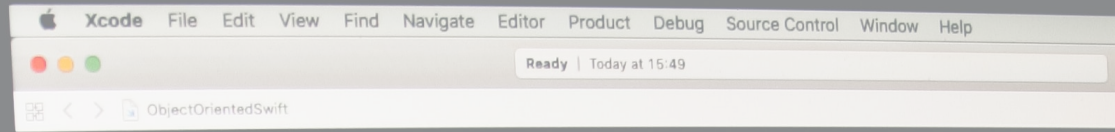
21 NOVIEMBRE 2021

PROGRAMACIÓN DE ESTRUCTURAS DE DATOS Y  
ALGORITMOS FUNDAMENTALES (GPO 11)

# REFLEXIÓN

# ACTIVIDAD

## 4.3



# Índice

<b>Planteamiento del problema</b>	<b>2</b>
<b>Estructura de Clases</b>	<b>2</b>
<b>¿En que consiste un grafo?</b>	<b>3</b>
<b>Casos de prueba y anexos</b>	<b>4</b>
<b>Referencias</b>	<b>5</b>

# PLANTEAMIENTO DEL PROBLEMA

Nuestro programa debe ser capaz de cumplir con los siguientes requisitos:

1. Abra el archivo de entrada llamado "bitacora.txt" lealo y almacene los datos en una lista de adyacencia organizada por dirección de ip origen.
2. Determine el fan-out de cada nodo
3. ¿Qué nodo tienen el mayor fan-out?

# ESTRUCTURA DE CLASES

Para la realización de este programa consideramos 3 clases:

La clase Graph que representa nuestro grafo, esta recibe en su constructor el nombre del archivo a leer (bitacora.txt) y en base a este genera una lista de adyacencia la cual se almacena como su atributo.

La clase IpAddress representa los primeros 3 dígitos de una dirección IP y cuenta con otro atributo que es el número de veces que se repite. Además cuenta con algunas sobrecargas de operadores para hacer que procesos como imprimir e eliminar duplicados sean más sencillos.

Por último tenemos la clase Algoritmos que cuenta con el método de ordenamiento mergesort adaptado para funcionar con objetos IpAddress.

## ¿EN QUE CONSISTE UN GRAFO?

Los grafos se conocen por tener conjuntos de vértices o nodos, usualmente con el símbolo  $V$ , mientras que  $A$  son los pares de vértices, que suelen denominarse al igual como arcos o ejes, siendo entonces el grafo como  $G = (V, A)$ . Los vértices en muchas situaciones de la vida real suelen verse como los objetos mientras que los arcos se tratan de las relaciones de estos mismos. Estos grafos se pueden representar de diferentes maneras tales como las matrices, las listas o bien las matrices dispersas.

En el caso de esta actividad, se ocupa la lista de adyacencia cuyo uno de sus beneficios es evitar el uso de espacio muerto o bien dónde no existe ninguna clase de información, dónde se representan los vértices, aristas o arcos del grafo que se encuentra indizado por los números de los vértices. El grafo que se ocupa se trata de un no dirigido puesto que no se encuentran en orden los pares de vértices que representan el arco además de que representa una relación simétrica debido a sus aristas, a diferencia de su contrario que es el grafo dirigido.

Los recorridos son maneras de llegar a todos los nodos que conforman al grafo desde su nodo de salida o de raíz, en los cuáles existen los de anchura y profundidad. Breadth First Search, por sus iniciales conocido como BFS o en español recorrido de anchura, se trata de una manera de en la que se realiza el recorrido de un grafo mediante niveles por los nodos que existen, de manera que sea aquellos nodos a un arco del nodo raíz, después a los que están a dos arcos del nodo raíz y así consecutivamente hasta llegar a todos los nodos que estén al alcance del nodo de salida.

Una de las interrogantes que se plantean en la actividad es sobre el Fan-Out en los nodos. Ya que existen diferentes maneras de interpretar lo que es el término de Fan-Out en el software, se decidió en este caso que el concepto se aplicaría a la cantidad de veces que se repiten los tres primeros números que conforman las IPs.

## CASOS DE PRUEBA Y ANEXOS

Puesto que las impresiones de nuestro programa son bastante extensas, ya que se imprime el fan-out de cada nodo, los casos de prueba completos serán anexos en el documento casos.txt.

```
Inicio de IP: 1 Repeticiones: 23
Inicio de IP: 2 Repeticiones: 14
Inicio de IP: 3 Repeticiones: 16
Inicio de IP: 4 Repeticiones: 12
Inicio de IP: 5 Repeticiones: 16
....
Inicio de IP: 995 Repeticiones: 14
Inicio de IP: 996 Repeticiones: 14
Inicio de IP: 997 Repeticiones: 17

Elemento con más repeticiones:
Inicio de IP: 708 Repeticiones: 32

--- BFS ---
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27
28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52
53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77
78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100 101
102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120
121 122 123 124 125 126 127 128 129 130 131 132 133 134 135 136 137 138 139
.....
```

Fragmento de Casos de Prueba

Como se puede observar en este pequeño fragmento el nodo con mayor fan-out (repeticiones) es el 708.

Además de anexa en el archivo “Lista de Adyacencia.csv” la lista de adyacencia que representa nuestro grafo y por ende nuestra bitácora.

## REFERENCIAS

Universidad de Valencia (2001). Estructura de Datos. Recuperado el 21 de noviembre de 2021 de [http://informatica.uv.es/iiguia/AED/oldwww/2001\\_02/Teoria/Tema\\_15alfa.pdf](http://informatica.uv.es/iiguia/AED/oldwww/2001_02/Teoria/Tema_15alfa.pdf)