

Reporte práctica 2.1

Planteamiento del problema

En esta práctica, se espera obtener del grafo obtenido de la página de facebook de Vox cuál es el hijo del nodo Vox, que tiene más hijos.

Procedimiento

Para obtener el grafo de la página de Vox, utilicé una herramienta llamada *Netvizz*, colocando la profundidad en dos para así obtener todas las conexiones que tienen las conexiones hacia Vox.

Una vez obtenido el grafo, se exportaron dos archivos que el programa *Gephi* nos permite analizar mediante *Data Laboratory*; el primero, toda la relación de los nodos del grafo en formato .csv, y el segundo todas las aristas del grafo también en formato .csv.

Desarrollo

Se desarrolló un código en python que permitió organizar los datos de los archivos .csv con la librería de *pandas* en un grafo. Al inicializar el grafo, este es de tamaño cero y tiene como nombre Vox Graph. Después, se procede a leer los archivos mediante .csv, donde mediante cada iteración, se irá agregando la información de los nodos con su respectivo id, nombre, nivel, y una lista con las conexiones, que en un principio será cero. Después, en la siguiente iteración donde se lee el archivo de las aristas, se procede a realizar las conexiones y a llenar la lista con las conexiones.

Una vez hecho eso, se procede a realizar el BFS y el DFS.

Resultados

Una vez realizado ambos métodos, llegué a la conclusión de que en este caso (ojalá haya estado bien implementado, jeje) de que me era más útil para resolver la pregunta el DFS, ya que el BFS sólo me mostraba los nodos en segundo nivel.

BFS

```
23649167822693)
BFS
0 Vox
1 Alvin Chang
1 David Roberts
1 Phil Edwards
1 Carlos Maza
1 Johnny Harris
231 Sarah Kliff
051 Vox Creative
1 Curbed
1 Racked
1 Eater
1 Polygon
1 The Verge
1 SB Nation
1 Vox Media, Inc
1 Ezra Klein
1 Matthew Yglesias
2 Grist.org
2 Cinematography Database
2 Slate.com
2 Atlas Obscura
2 Neatorama
2 Mental Floss
2 Francesca Fiorentini
2 Liz Plank
2 Architizer
2 Curbed Philly
2 Aamodt / Plumb
2 Who What Wear
2 Public School
```

DFS

```
0 Vox
1 Matthew Yglesias
2 Ezra Klein
3 Eater
3 SB Nation
23649164 SB Nation Arizona
051: 5 AZ Snakepit
5 Five for Howling
6 Broad Street Hockey: For Philadelphia Flyers Fans
6 Fear The Fin
6 Mile High Hockey
7 Puck Daddy
8 Dr. Saturday
9 Yahoo Sports NBA
10 theScore NBA
8 Yahoo Sports
6 Raw Charge
7 Second City Hockey
7 The Daily Stampede - For USF Fans
7 Copper & Blue: For Edmonton Oilers Fans
8 Nucks Misconduct
9 Battle of California
9 Stanley Cup of Chowder
10 Canes Country
10 Hockey Wilderness
11 Matchsticks & Gasoline: For Calgary Flames Fans
12 Die by the Blade
12 Blueshirt Banter: For New York Rangers Fans
12 Defending Big D
```

Conclusiones

Por los resultados, se puede concluir que el hijo que cumple con la mayor cantidad de hijos es la página Mathew Yglesias utilizando el algoritmo DFS, ya que con el BFS, no se puede ver con claridad.

La ventaja que pude observar al utilizar el DFS en este caso, es que literal me fue proporcionando todos los nodos hijos de los grafos uno por uno, mientras que en la visualización del BFS, lo va haciendo por niveles lo que me impide contestar esta pregunta con mayor facilidad.

Creo yo, que las ventajas y desventajas de cada uno depende para el uso y la solución a la cuál se quiera llegar, ya que habrá casos en los que convenga utilizar uno, y habrá otras en los que convendrá utilizar la otra.

Respecto a la parte de la limitación de la búsqueda, sí se puede limitar poniendo un límite en el contador de los ciclos en los respectivos algoritmos para solamente limitar la profundidad del cuál se hará la búsqueda; sin embargo, puede discrepar ahí los datos, ya que cada algoritmo tiene un método distinto de búsqueda.

1ra iteración 2da iteración