# INFORME DEL TRABAJO DE INSERCIÓN PROFESIONAL

### 1. Introducción

Este trabajo surge de la necesidad de verificar la veracidad del algoritmo de H-grafo descrita por Min Chih Lin, Francisco J. Soulignac y Jayme L. Szwarcfiter en su artículo "Arboricity, h-index, and dynamic algorithms" la cual se elabora en base a la estructura de grafo. Dicha estructura es simple, consistente de tres elementos clave por cada uno de los vertices del grafo. La implementación incluye las operaciones básicas relacionadas a las aplicaciones dinámicas tales como la inserción y borrado de vertices o aristas junto con funciones de consulta sobre el grafo. El objetivo principal es demostrar la gran diferencia en cuanto a eficiencia del algoritmo comparado con otras implementaciones más conocidas. Con este fin, hemos elaborado algunos casos de ejemplo hacia el final del trabajo con los que se pueden apreciar estas distinciones.

### Borrado de arista vw

* Primero se asume que el d(v) <= d(w)
* Entonces el neighbor\_w esta en HighNeiborhood de v (H(v))

Vértice v

[ […],[...], [...w...]]

Neighbor

\*neighbor ->puntero al vertice

\*list\_pointer ->puntero a la lista donde

esta v en w

\*self\_pointer → puntero directo a v en w

* Se itera hasta H(v) hasta encontrar a neighbor\_w.
* Luego, se utilizan los punteros de neighbor\_w para ubicar y borrar del vecindario de N(w) a v.

Actualizar el vecindario

\* Se itera el H(v), y se actualiza el vecindario de cada neighbor moviendo a v desde la lista N(d(v)) hacia N(d(v) -1)

vértice v vértice z

[ […],[...], […z..]] [ […],[.V.], […]]

* Se actualizan los punteros de z.
* Después se unen el N(v, d(v)-1) con el HighNeiborhood de v.

[ […],[A, B] ++ […z..]] → [ […],[A,B…z..]]

Por último se repite el proceso con el vértice w.