Diana Isabel Muñoz Castillo

**Proyecto en Python**

*Resumen:*

En esta práctica se van a implementar tablas de hash en Python. Las implementaciones deben tomar en cuenta todos los principios del software de calidad: eficiencia, robustez, generalidad, modularidad, legibilidad, etc. Los aspectos relevantes acerca del funcionamiento de las tablas de hash se presentan abajo de manera abstracta (es decir, independiente del lenguaje de implementación). Uno de los puntos importantes a considerar cuando se utilizan las tablas de hash, mencionado en dicha discusión teórica, es el manejo de colisiones. Para esta práctica se tendrá que utilizar el método conocido como "encadenamiento separado”. Otro punto importante para considerar es la función de hash que se va a utilizar. Cuál es la más apropiada depende de las características de los datos que se van a almacenar en la tabla de hash. Después de tener implementadas las tablas de hash se debe diseñar y realizar una serie de experimentos que permita medir tiempos de ejecución (y cualquier otro parámetro relevante) para obtener un entendimiento detallado del funcionamiento de las tablas de hash bajo la estrategia de encadenamiento separado para el manejo de colisiones. Algunos parámetros que deben variar (deben estar contemplados dentro del diseño de la serie de experimentos) para obtener suficientes datos para poder llegar a conclusiones generales son: (a) la cantidad de datos con los que se alimente la tabla de hash, n, en relación con el tamaño de la misma, m (n mucho mayor a m vs. n aproximadamente igual a m vs. n mucho menor que m), (b) el tipo de operación a realizar sobre la tabla de hash (inserción de datos vs. búsqueda de datos vs. eliminación de datos), y (c) cualquier otro que se les ocurra que permita explorar algún aspecto de las tablas de hash y/o el método de encadenamiento separado para el manejo de colisiones.