

OBSERVACIONES DEL LA PRACTICA

María Alméciga 202023369

Andrés Vargas 202013817

Preguntas de análisis

- 1) ¿Qué relación encuentra entre el número de elementos en el árbol y la altura del árbol?

Ya que se están utilizando arboles BST y no se especifica si están balanceados o no, se realiza la proporción para determinar entre que rango estaría balanceado o no el árbol. Cuando llega al caso de balanceo con $N=1177$, se determina a partir de $\log_2(N)$, y esto se aproxima hacia arriba, por lo que será 11. En su peor caso debido a su cantidad de llaves será $N-1$, por lo que sería 1176. La altura del árbol puede ser aleatoria pero la relación más cercana sería que está casi balanceado el árbol ya que 29 está más cerca de 11 que de 1177. Esto llevaría a decir que es posible determinar con los niveles de altura la cantidad de niveles que puede haber. Esto se muestra a partir de 2^n , donde n son los niveles del árbol y el resultado es la cantidad de elemento que puede tener el árbol.

- 2) ¿Si tuviera que responder esa misma consulta y la información estuviera en tablas de hash y no en un BST, cree que el tiempo de respuesta sería mayor o menor? ¿Por qué?

La implementación llevaría más tiempo ya que, a pesar de que los elementos pudieran estar indexados de la misma forma que en el árbol (con las fechas de los crímenes) la búsqueda no resultaría tan simple, pues tendría que recorrer todas las llaves para encontrar el las que están en el rango indicado. Por otro lado, en el árbol, ya que los elementos pueden estar ordenados por sus llaves, la búsqueda implicaría solamente encontrar la primera fecha dentro del rango, y llegar hasta la última indicada, pues se garantiza que están en orden y no se excluirán otras necesarias.

- 3) ¿Qué operación del TAD se utiliza para retornar una lista con la información encontrada en un rango de fechas?

Se utiliza la operación `values()`, que “retorna todos los valores del árbol que se encuentren entre $[keylo, keyhi]$ ”, pues en efecto a esta función se le están pasando la fecha menor y la mayor entre las que se van a buscar los datos, y nos devuelve la información encontrada en el rango.

```

def values(bst, keylo, keyhi):
    """
    Retorna todas los valores del arbol que se encuentren entre
    [keylo, keyhi]

    Args:
        bst: La tabla de simbolos
        keylo: límite inferior
        keylohi: límite superiorr
    Returns:
        Las llaves en el rango especificado
    Raises:
        Exception
    """
    try:
        lstvalues = lt.newList('SINGLE_LINKED', bst['cmpfunction'])
        lstvalues = valuesRange(bst['root'], keylo, keyhi, lstvalues,
                                bst['cmpfunction'])
        return lstvalues
    except Exception as exp:
        error.reraise(exp, 'BST:Values')

```