**Rta 2:**

Big tree es una librería de Python usada para la implementación de arboles y gráficos, que a diferencia de árboles binarios tradicionales, Big tree permite múltiples hijos por nodo en los árboles, lo que lo hace útil para representar estructuras no binarias, como categorías de productos, archivos, organizaciones, etc.

El siguiente ejemplo es una implementación correcta de un árbol no binario:

[**https://www.programiz.com/online-compiler/8kNuq9acuAszh**](https://www.programiz.com/online-compiler/8kNuq9acuAszh)

**Rta 3:**

Los árboles de Huffman son una estructura usada para comprimir datos de forma eficiente, un ejemplo de codificación de Huffman, la historia de la codificación Huffman se remonta al año 1952 cuando su inventor, David A. Huffman, empezó el desarrollo de su tesis de grado en el Instituto Tecnológico de Massachusetts, cuya premisa dada por su docente fue “Encuentra el método más eficiente para comprimir datos con codificación binaria, o escribe un trabajo de investigación sobre el tema”, de este trabajo nació el Algoritmo de Huffman, que se encarga de construir un árbol binario basado en la frecuencia de aparición de los símbolos o caracteres en un conjunto de datos. El algoritmo asigna códigos binarios más cortos a los símbolos más frecuentes y códigos más largos a los menos frecuentes, logrando así una compresión sin pérdida de información.

El funcionamiento del algoritmo comienza agrupando los caracteres en nodos hoja con sus respectivas frecuencias. Luego, se combinan de forma iterativa los dos nodos con menor frecuencia para formar un nuevo nodo padre, cuya frecuencia es la suma de los dos nodos hijos. Este proceso se repite hasta que solo queda un nodo, que se convierte en la raíz del árbol de Huffman. El recorrido del árbol desde la raíz hasta cada hoja define el código binario de cada carácter.

Este método se utiliza ampliamente en algoritmos de compresión como los de archivos .zip, imágenes .png y estándares de transmisión de datos como MP3 y JPEG. Su eficiencia se debe a que reduce el número total de bits necesarios para representar un mensaje, especialmente cuando ciertos caracteres aparecen con mayor frecuencia que otros.

**HECHO POR:** David Parra (2230047) y Julian David Vargas Gomez (2221885)