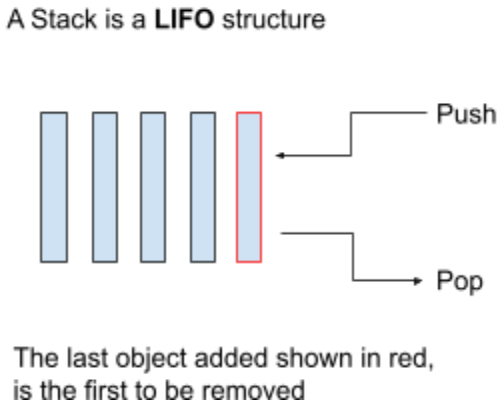
1. The deque module is part of the ”collections” python library

2. The two differences that distinguish a tree from a graph are:
    a. Trees have a hierarchical structure - with one node at the root, but graphs do not.
    b. Graphs can have more than one path between two vertices, but trees can only have one path between two vertices.

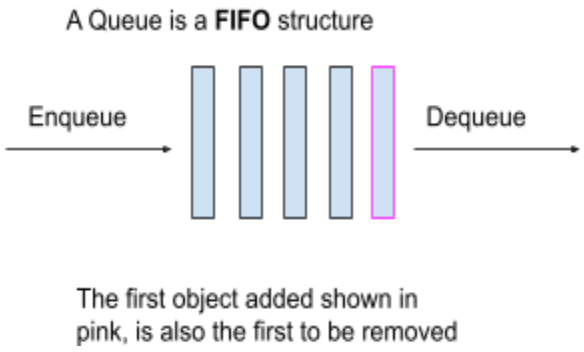
3. The definitions of time and space complexity are:
    a. Time complexity is a measure of how quickly an algorithm runs, as a function of the size of its inputs.
    b. Space complexity is a measure of how much auxiliary memory an algorithm takes up, also with respect to input size.
    Both are expressed using Big O notation, with variables (e.g. N & M) to represent inputs to the algorithm.

4. The bubble sort algorithm moves through a list and compares pairs of elements that are next to each other (adjacent). It then swaps adjacent elements in a list if they are in the wrong order. This is done repeatedly, passing through the list until it is all sorted. Since it has to go through the unsorted part of the list each time, the worst case time complexity for this algorithm is O(n2). Its space complexity is O(1). What is guaranteed at the end of the first pass is that the last element will be the largest one.

5. LIFO and FIFO are acronyms which describe the principles by which objects are inserted and removed in data structures. Last-In First-Out (LIFO) is where the last object to be inserted into the structure is the first one to be removed. A Stack is a LIFO data structure.



First-In First-Out is where the first object inserted into the structure is the first one to be removed. A Queue is a FIFO data structure.



Note: Both diagrams were made by me, inspired by class material.

6. A Balanced Binary Tree is where the difference in the height of the left and right subtrees of every node is no more than 1. The best root is the median value. To search using a Balanced Binary Tree:
    a. If the data is less than the key value, search the the left of the tree
    b. If the data is greater than or equal to the key value, search the right subtree
    c. Repeat for every node

    The time complexity for searching a Balanced Binary Tree is O(log(n)).