

* Describe Python's built-in data structure?

We can divide Python data structures mainly into three categories:

- numeric
- sequence
- mapping

But there are also other data types like classes, files etc.

These are some of the data types and categories they are in.

- None - Null object
- Numeric - integer, float, complex, boolean
- Sequences - string, list, tuple, range
- Mapping – dictionary, set, frozenset

* Describe the Python user data structure?

There are some data structures that are user defined. These data structures are generally imported from collections library and these include:

- Stacks - Apply LIFO structure - last in first out
- Queue - Apply FIFO structure - first in first out
- Linked list - Dynamic array - has pointers pointing to the next element of list
- Tree - A nonlinear hierarchical data structure that consists of nodes
- Graph - A nonlinear data structure consisting of nodes and edges

* Describe the stages involved in writing an algorithm?

We can say that algorithms are composed of these 4 stages

- Sequential operations
- Actions based on the state of a data structure
- Iteration, repeating an action a number of times
- Recursion, calling itself on a subset of inputs

* Outline the components of a good algorithm?

- A good algorithm should produce the correct output for every legitimate input.
- A good algorithm should be executed efficiently in as few steps as possible.
- A good algorithm should be designed so that others can understand it and modify it to specify a solution to the additional problem.

*Describe the Tree traversal method?

Basically tree traversing is processing the data of a node once in some order in a tree.

And there are two main types of tree traversal method.

- Breadth first traversal
- Depth first traversal

Breadth first is level order traversal. It traverses nodes in layers. It means that we divide our tree into portions at each branching and take the nodes in same height as same group.

Depth first search has 3 types

- Preorder
- Inorder
- Postorder

* Explain the difference between inorder and postorder tree traversal?

Inorder traversal- For inorder traversal we start from top node and travel through the left subtree until there is no node on the left side. We return last left node in subtree. After we should go back to the parent of last node and add it to the traversal order. We now check right subtree until last right node and again do same process for right subtree as we did in left subtree.

Postorder traversal - Again we start from root node going until leftmost node. We can add first node to postorder traversal after finding left node with neither right nor left node. Then we return back to the parent of this left node and look at its right side. We look for the node which is the last node and add it to

the traversal list. then we return back to parent node and do same process until all of the nodes are added to the traversal list.

Main difference is in Inorder traversal we add the parent nodes of leftmost node on the returning way, however in postorder we traverse first left subtree then right subtree and root finally. Inorder traversal structure

Left subtree - Root - Right subtree

Postorder traversal

Left subtree - Right subtree - Root