**Assignment 1**

1. List, Tuple, Set and Dictionary are the python built-in data structures.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Features / Data structures | List | Tuple | Set | Dictionary |
| 1. Collection type | Ordered collection of items | Ordered collection of items | Unique elements do not follow order | Ordered collection of items |
| 1. Nesting | Can be nested | Can be nested | Cannot be nested | Can be nested |
| 1. Mutable or not | Mutable | Immutable | Mutable | Mutable |
| 1. Duplicable | Allow duplicate elements | Allow duplicate elements | Not allow duplicate elements | Not allow duplicate elements |
| 1. Syntax | [ ] | ( ) | { } | {key: value} |
| 1. Use cases | 1. Purpose of when items get modified (Adding, Deleting, Reading) 2. Mostly used due to subsequent functions of list | 1. Use less memory,  2. Faster execution,  3. Prevent accidental modification of data | 1. Existence/membership test | 1. Improves readability  2. Easier to debug when code becomes more streamed  3. Useful when manual indexing needed |

1. Data structures that aren’t supported by python but can be programmed to reflect the same functionality using concepts supported by python are user-defined data structures.

Eg: -

1. Array
2. Linked list
3. Queue
4. Stack
5. Tree
6. Graph
7. Hashmap
8. Step 1: Obtain a description of the problem.

Step 2: Analyze the problem.

Step 3: Develop a high-level algorithm.

Step 4: Refine the algorithm by adding more detail.

E.g. : - Algorithm for summing two numbers

1. Input specified: 0 / more well-defined inputs,

Output specified: 1 / more well-defined outputs,

Definiteness: Same output for same inputs at any time,

Finiteness: Finite number of instruction execution at any time,

Independent,

Effectiveness,

Each instruction/ execution should be crystal clear.

1. A tree traversal is a method of visiting every node in the tree. Traversal can happen in depth-first order or breadth-first order. Pre-order, in-order and post-order are some of the three methods to traverse a tree in a depth-first order manner and level order is a breadth-first order method.

Tree traversal is used to list down all the elements in a tree or search for an element in the tree whether it is in or not, if it is in then what is the position.

1. In the in-order traversal, right sub tree will be visited after root node but it is opposite in post-order traversal

In-order traversal:

1. Traverse the left sub tree
2. Visit the root
3. Traverse the right sub tree

Post-order traversal:

1. Traverse the left sub tree
2. Traverse the right sub tree
3. Visit the root