**Data Structures**

Name

Institutional Affiliation

Course

Professor

Date

**Data structures**

Data structures are a specialized way to organize and store data in computers so that we can perform operations on the stored data more efficiently. Data structures are used in almost all applications and software that have been developed (McKinney, 2010). The way of using these data structures varies from application to application.

There are different data structure ways of organizing and storing data.

* Arrays
* List
* Dictionaries etc

**Arrays**

Arrays have various uses across different algorithms and are available in almost all programming languages, including python. More than one data item can be stored in an array but with the same data type. Items in an array are allocated adjacent memory locations called elements (Pivarski et al., 2020).

The total number of items or elements in the array is called array length. Accessing array items can be done through its position called an index. An index of an array will always start at 0, which means if there are n elements, the index at some position p will be p-1.

An array is a contiguous data structure since they store information in adjoining memory blocks.

**Lists**

A list is an ordered collection of items, one of the essential data structures in python. Lists are implemented as dynamic arrays, meaning that their size can be dynamically modified at runtime. Creating a list in python means creating a python object of a specific type. All items in a list are put in square brackets and separated by commas. Example

Items = [item1, item2, item3, item\_n]

A list in python can contain another list or a sub-list, which can also contain another list in itself. Lists in python are mutable, meaning they can be altered even after being created. Operations performed on the lists are:

* Add
* Search
* Shift
* Move
* Delete

**Justification for Lists and arrays**

Arrays are one of the best ways to organize and store data and can be manipulated at any given instance (Davydov et al., 2020). Unlike other data structure methods, arrays can store many values of data of the same data type. They are flexible data structures to work within python. There are two reasons why arrays are more preferred, which are:

* Less memory usage – Memory usage for this data structure is far less than that of other data structures
* Less execution time – It takes less time to complete the execution of an array.

A linked list is also preferred for data structures since it is a dynamic structure that can grow and shrink at runtime. Lists also use efficient memory since the linked list's size increases or decreases at runtime; thus, memory is allocated only when needed. The insertion and deletion tasks are also more manageable in the list since there is no need to shift elements after the insertion or deletion of an element. Only the address present in the next pointer needs to be updated.

**References**

Davydov, V. V., Nikolaev, D. I., Bukharov, G. D., & Pavlova, Z. S. (2020). Methods for protecting of diode arrays during laser pumping. 2020 IEEE International Conference on Electrical Engineering and Photonics (EExPolytech). <https://doi.org/10.1109/eexpolytech50912.2020.9243948>

McKinney, W. (2010). Data structures for statistical computing in Python. Proceedings of the 9th Python in Science Conference. <http://conference.scipy.org/proceedings/scipy2010/pdfs/mckinney.pdf>

Pivarski, J., Elmer, P., & Lange, D. (2020). Awkward arrays in Python, C++, and Numba. EPJ Web of Conferences, 245, 05023. <https://doi.org/10.1051/epjconf/202024505023>