MYO THIDA

ALGORITHMS & DATA STRUCTURE

Myo Thida

WHAT IS DATA STRUCTURE?

COMPUTER MEMORY

$$
\begin{aligned}
00000000 &= 0 \\
00000001 &= 1 \\
00101001 &= 41 \\
10000000 &= 128 \\
11111111 &= 255
\end{aligned}
$$

- **The program and its data are loaded from the storage device (like an SSD or HDD) into RAM.**
- **The CPU (Central Processing Unit) processes the data in RAM as it executes the instructions provided by the program.**
- **Any changes made to the data during the program's execution may be temporarily stored in RAM.**
- **Data is saved in binary format.**

Classification of Data Structure

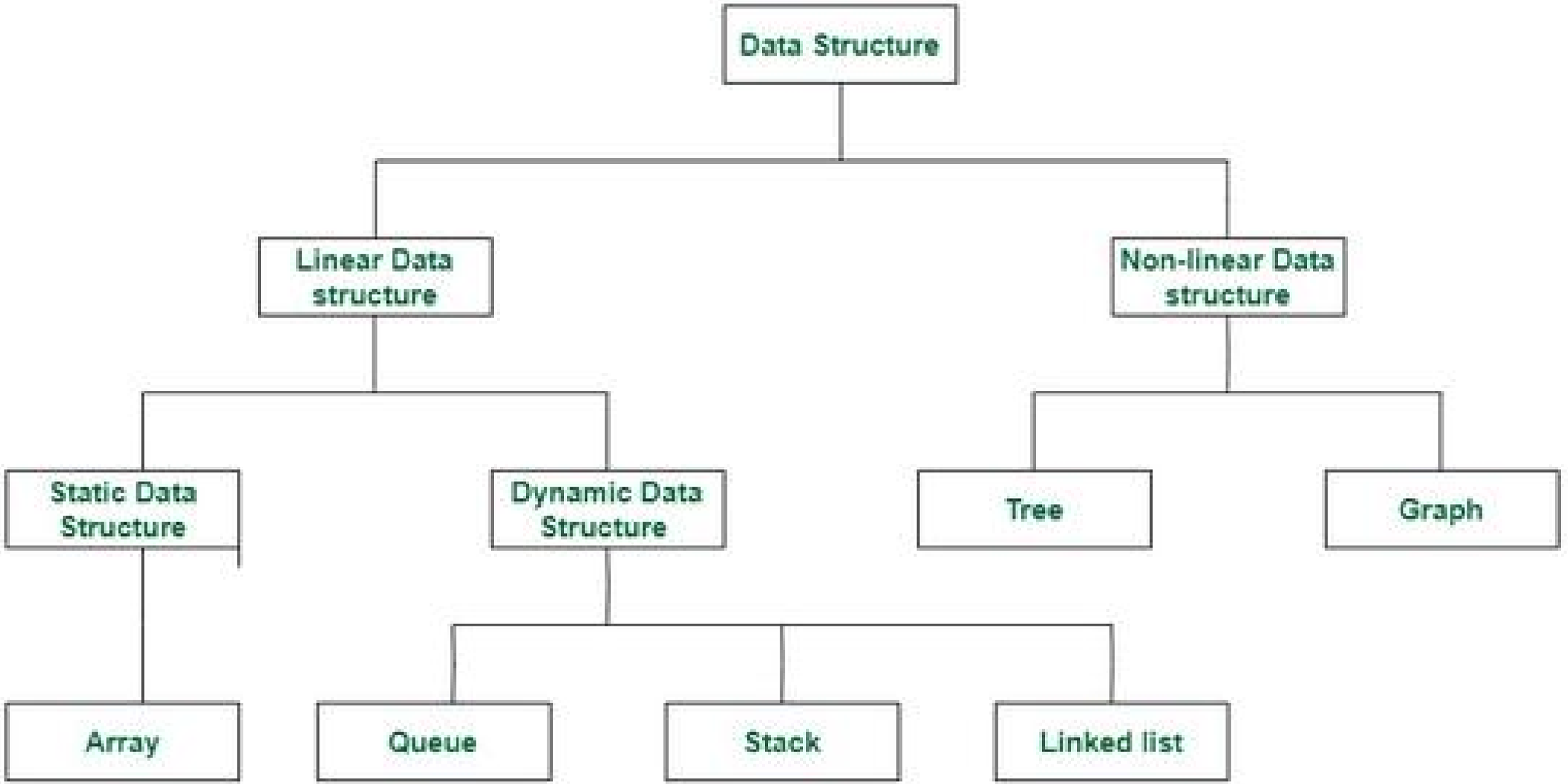

Image Credit to https://www.geeksforgeeks.org/

ARRAYS

- Most popular data structure.
- A linear data structure that collects elements of the same data type and
- Stores them in contiguous and adjacent memory locations.

	Memory Location	Value
0	0x2b4aa796b10	16
1	0x2b4aa796c10	24
2	0x2b4aa796d90	36
3	0x2b4aa796e90	44
4	0x2b4aa796f90	52
5	0x2b4aa796c50	26
6	0x2b4aa796db0	37
7	0x2b4aa796d50	34
8	0x2b4aa7c5050	56
9	0x2b4aa7c5310	78

MY_ARRAY = [16, 24, 36, 44, 52, 26, 37, 34, 56, 78]

ARRAYS

- Arrays work on an index system starting from 0 to (n-1), where n is the size of the array.

MY_ARRAY [0]?

MY_ARRAY[4]?

MY_ARRAY[10]?

	Memory Location	Value
0	0x2b4aa796b10	16
1	0x2b4aa796c10	24
2	0x2b4aa796d90	36
3	0x2b4aa796e90	44
4	0x2b4aa796f90	52
5	0x2b4aa796c50	26
6	0x2b4aa796db0	37
7	0x2b4aa796d50	34
8	0x2b4aa7c5050	56
9	0x2b4aa7c5310	78

ARRAYS IN PYTHON

- **No built-in data type**
- **Need to import using array or numpy.**
- **Need to enclose using .array function.**
- **Contain elements of the same data types.**
- **Can apply direct arithmetic operations.**

```
my_array = np.array([16, 24, 36, 44, 52, 26, 37, 34, 56, 78])
```

2D ARRAYS IN PYTHON

a = np.array([[1, 2, 3, 4], [5, 6, 7, 8], [9, 10, 11, 12]])

```
array([[ 1,  2,  3,  4],
       [ 5,  6,  7,  8],
       [ 9, 10, 11, 12]])
```

https://numpy.org/doc/stable/user/absolute_beginners.html

2D ARRAYS IN PYTHON

a = np.array([[1, 2, 3, 4], [5, 6, 7, 8], [9, 10, 11, 12]])

a[0,3]?

a[2, 3]?

a[1, 4]?

https://numpy.org/doc/stable/user/absolute_beginners.html

ARRAY FUNCTIONS

- np.array(),
- np.zeros(),
- np.ones(),
- np.empty(),
- np.arange(),
- np.linspace()

```
np.zeros(3)
```
✓ 0.0s

```
array([0., 0., 0.])
```

```
np.ones(4)
```
✓ 0.0s

```
array([1., 1., 1., 1.])
```

```
np.arange(5)
```
✓ 0.0s

```
array([0, 1, 2, 3, 4])
```

ARRAY FUNCTIONS

- **Generate a sequence of 21 evenly spaced numbers between 0 and 100.**

```
np.linspace(0,100, 21)
```

✓ 0.0s

```
array([  0.,   5.,  10.,  15.,  20.,  25.,  30.,  35.,  40.,  45.,  50.,
        55.,  60.,  65.,  70.,  75.,  80.,  85.,  90.,  95., 100.])
```

PYTHON BUILT-IN DATA TYPE LIST

PYTHON - LISTS

MY_LIST = [16, 24, 36, 'NAME', 52, 'AGE']

- **Built-in data type**
- **Enclose using square bracket []**
- **Can contain elements of the different data types.**
- **Cannot apply direct arithmetic operations.**

	Memory Location	Value
0	0x2b4aa796b10	16
1	0x2b4aa796c10	24
2	0x2b4aa796d90	36
3	0x2b4aa7c8130	name
4	0x2b4aa796f90	52
5	0x2b4c9c0e270	age

PYTHON BUILT-IN LISTS

MY_LIST = [16, 24, 36, 44, 52, 26, 37, 34, 56, 78]

	Memory Location	Value
0	0x2b4aa796b10	16
1	0x2b4aa796c10	24
2	0x2b4aa796d90	36
3	0x2b4aa796e90	44
4	0x2b4aa796f90	52
5	0x2b4aa796c50	26
6	0x2b4aa796db0	37
7	0x2b4aa796d50	34
8	0x2b4aa7c5050	56
9	0x2b4aa7c5310	78

procedure $max(a_1, a_2, \ldots, a_n$: integers)
$max := a_1$
for $i := 2$ **to** n
    if $max < a_i$ **then** $max := a_i$
return max {max is the largest element}

ARRAY & LIST EXERCISES

- **Write an algorithm to find the sum of all even numbers in an array without using built-in functions.**
  - **Instance: my_array = [3, 4, 5, 8, 13, 54]**
- **Write an algorithm to find the maximum element in an array without using built-in functions.**
  - **Instance: my_array = [3, 4, 5, 68, 13, 54])**
- **Write an algorithm to move all zeros to the end of the given array of integers while maintaining the order of non-zero elements.**

THANK YOU!

Feel free to approach me if you have any questions.