Week 1 - Arrays

Website

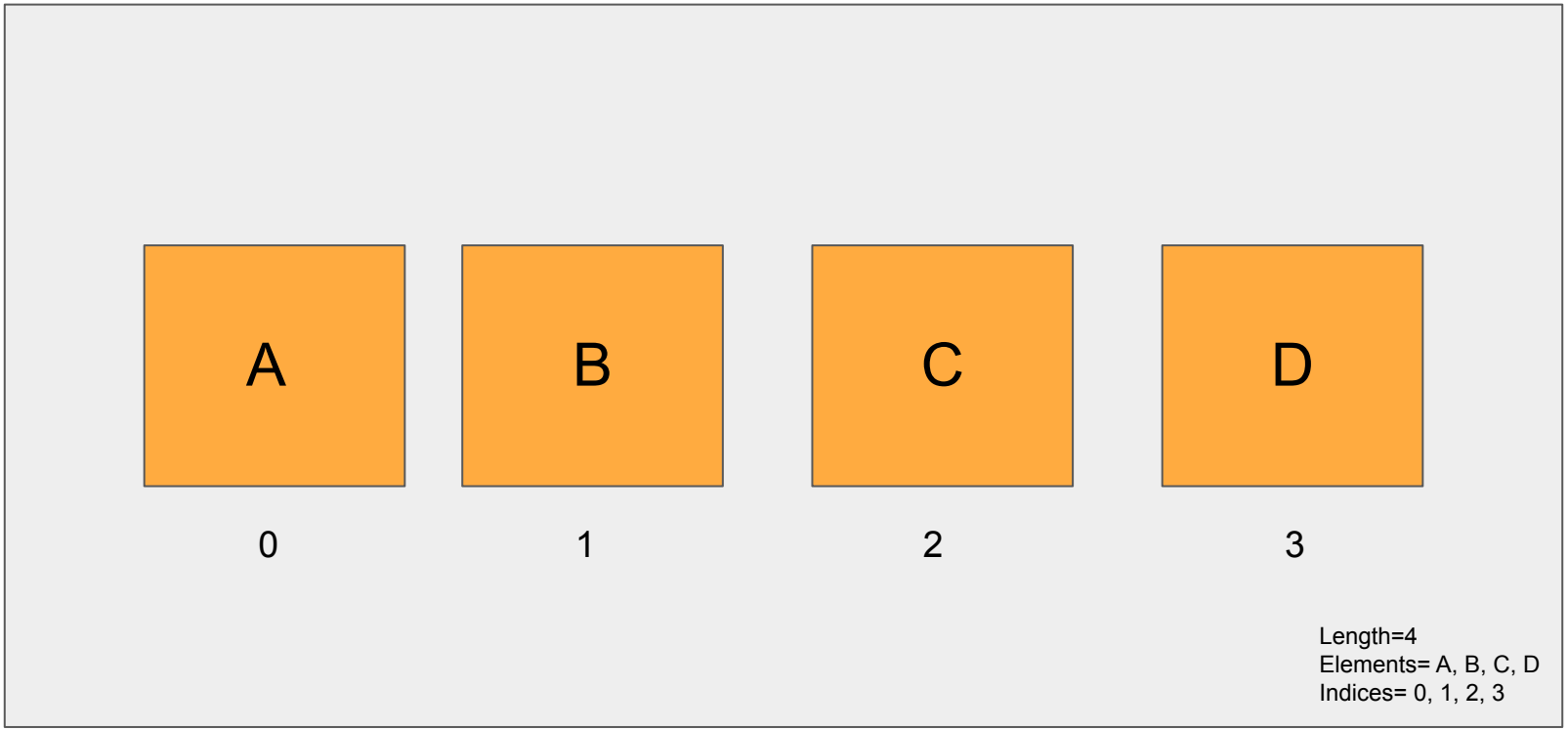
What are Arrays, Importance, Etc

- hold values of same type at neighboring memory locations
- concerned about the subscript/position/index and type of element itself
- total number of elements in array is called **length**
- Python (called lists), JavaScript, Ruby, PHP DON’T need a size defined before creating (usually easier for interviews)

Why We Need Arrays

- Storing many values in a single variable
- Better at processing many values easily and quickly
- Sorting and searching values easier

Array Diagram



Array Advantages

- Can store multiple elements of same type with unique variable name
- Access elements is fast as long as you have the index

Arrays Disadvantages

- Adding/removing elements from/into array is slow because they need to be shifted to accommodate the new/missing element (exception is at end of array)
- For fixed array sizes, if a new element exceeds the size, a new array must be initialized and all the elements from old array must be copied over

Fun Fact: the act of creating a new array and copying elements over takes **O(n) time**

Common terms with arrays

Subarrays: a range of neighboring values in an array

Array: [2, 3, 4, 7, 8, 9]

Subarray: [4,7,8]

Subsequence: a range of values in an array without changing order of elements

Array: [12, 13, 17, 8, 9]

Subsequence: [12, 8, 9]

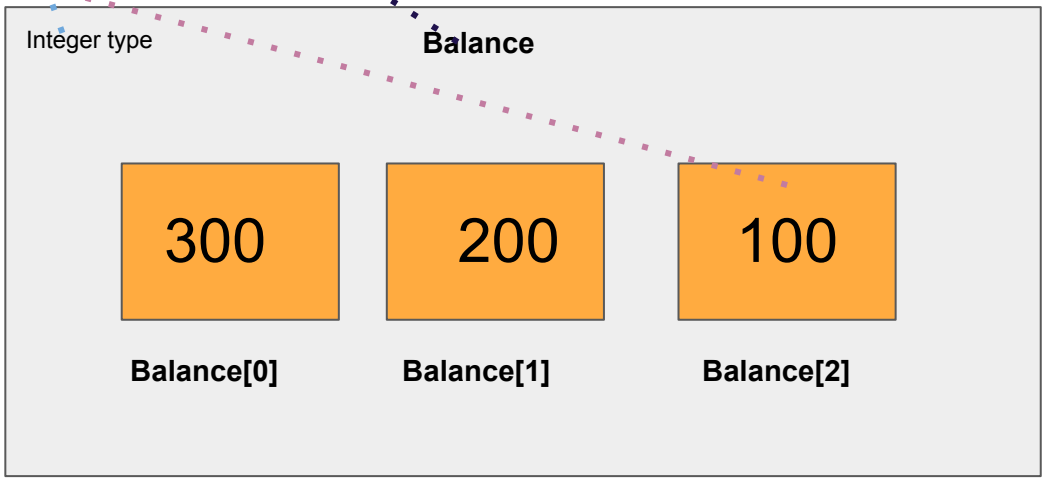
Time Complexity

Operation	Big O	Note
Access	O(1)	
Search	O(n)	
Search (sorted array)	O(log(n))	
Insert	O(n)	Elements shifted to right by 1
Insert (at end)	O(1)	No elements shifted
Remove	O(n)	Elements shifted to left by 1
Remove (at end)	O(1)	No elements shifted

3 Common Array Initializations for all Languages

- Array name
- Data Type
- Elements

Python Array Initialization Example:
balance = array.array('i', [300, 200, 100])



To Declare an Array in Python

typecode elements

```
balance = array.array('i', [300, 200, 100])
```

array name module method

Type Codes in Python

‘c’ = character

‘u’ = Unicode character

‘i’ = integer

‘f’ = float

‘l’ = long

Arrays Summary

- Contiguous (neighboring) area of memory with equal sized elements indexed by contiguous integers
- Constant time O(1) access to any element
- Constant time O(1) to add/remove element at end
- Linear time O(n) to add/remove at beginning or middle location

Array Python Documentation & Method Examples

Documentation

Method Examples

Sources

https://www.techinterviewhandbook.org/coding-interview-study-plan/

https://www.guru99.com/array-data-structure.html