

# DATA STRUCTURES



# Static Array

- A fixed sized array
- Once created size does not change
- Supports random access
- Very common in modern languages (C#, Java)
  - JavaScript does not support it directly
  - Can simulate
- No reallocation since add/remove are not available

# Dynamic Array (A.K.A ArrayList)

- An array that can change its size
- Supports add/remove
- Supports random access
- Might cause reallocation when adding/removing new items
- JavaScript supports that concept through the Array data type []

- Supports push & pop
- No random access (index based)
- A.K.A LIFO – Last in first out
- In JavaScript can be simulated using plain array

# Linked List

- Supports insertFirst, insertLast, removeFirst, removeLast
- Each node is linked to the next node
- Sometimes is implemented as doubly linked list
- No random access 😞
- No reallocation when adding new node 😊

# Binary Tree

- Supports add, remove
- Each node may have at most two children: left & right
  - Left child is smaller than parent
  - Right child is greater than parent
- Thus, data is always sorted
- No random access
- Efficient search  $O(\log N)$

# Hash Table

- An array where items are located according to their hash value
- Two distinct items might have the same hash value
  - Will be linked as the same location
- The hash function should avoid duplicates as much as possible
- Very efficient searching almost  $O(1)$