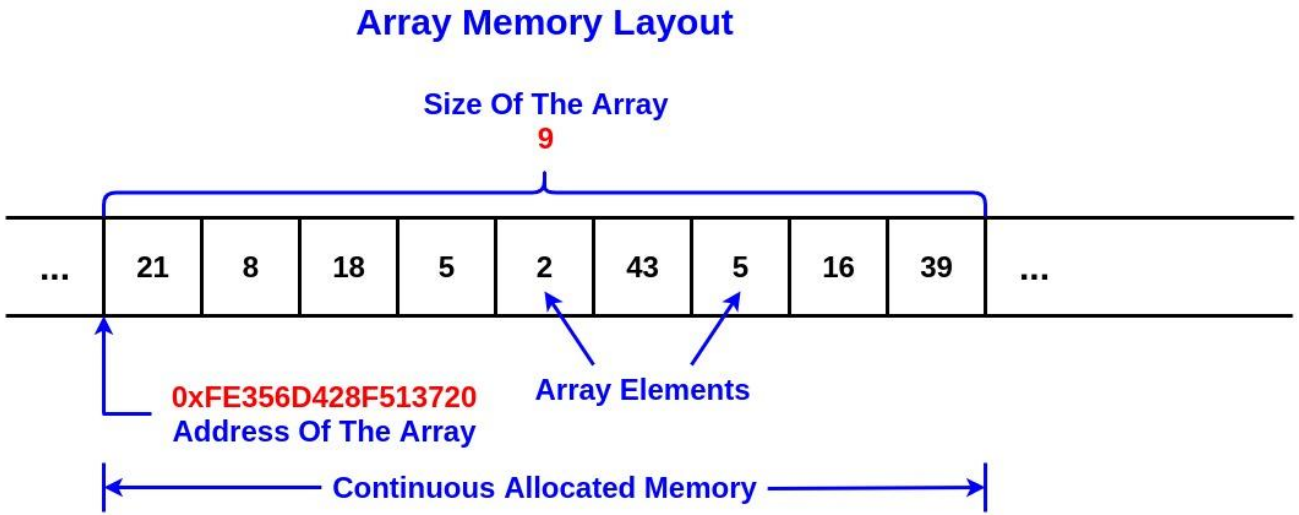
Arrays: Operations

Agenda

- What are arrays?
- How are arrays allocated?
- What operations can be performed on arrays?
- How are array operations implemented?
- Summary

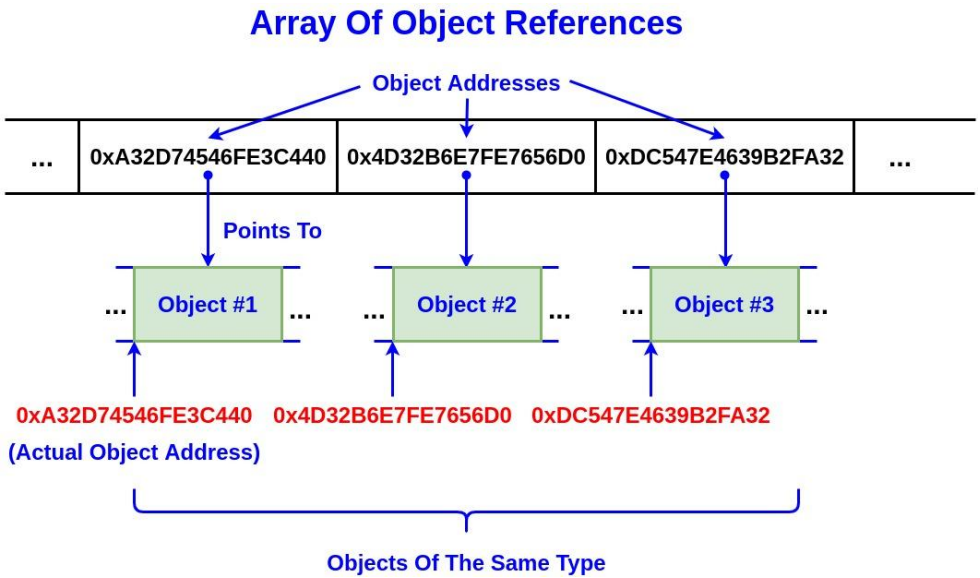
What is an array?

- An array:
  - is a basic data structure to organize a set of elements.
  - has a fixed size - at initial allocation.
  - has a hardware implementation support.



What is an array?

- Some more properties of an array:
  - Homogenous: holds elements of same type.

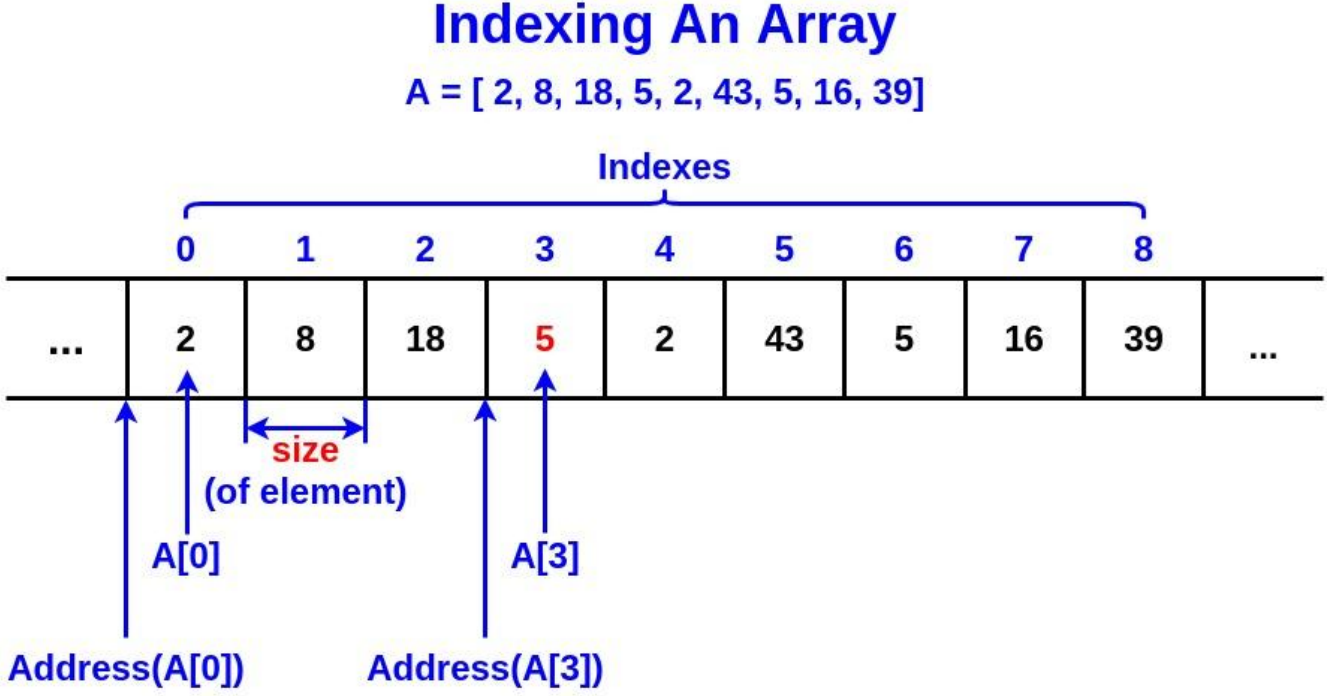


What is an array?

- Some more properties of an array:
  - Random access: Elements can be accessed directly
    - Constant-time operation
    - Made possible by Indexing

- Indexing
  - Directly access any element with the [] operator
  - Manipulates element addresses
  - Relies on array being homogeneous

Array operations: Indexing



Array operations: Indexing

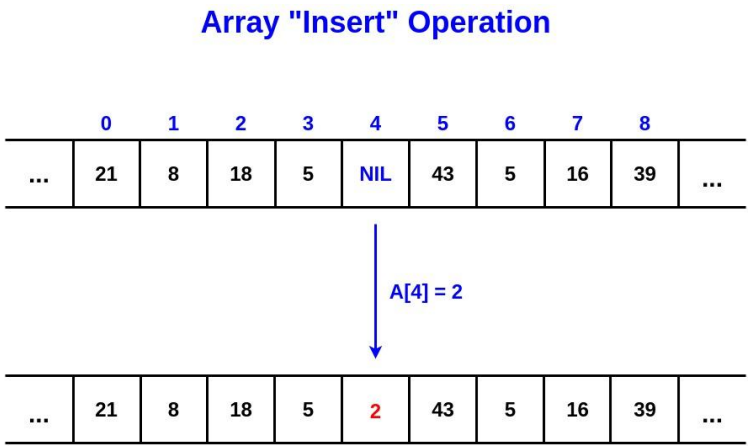
- How indexing works
  - Start address of the array: the base reference
  - Individual elements have addresses relative to base
  - Simple address arithmetic

Array operations: Indexing

- Indexing is a constant time operation
    - If an array was not homogeneous, indexing arithmetic would not make sense.
- Indexing is used by all other array operations.

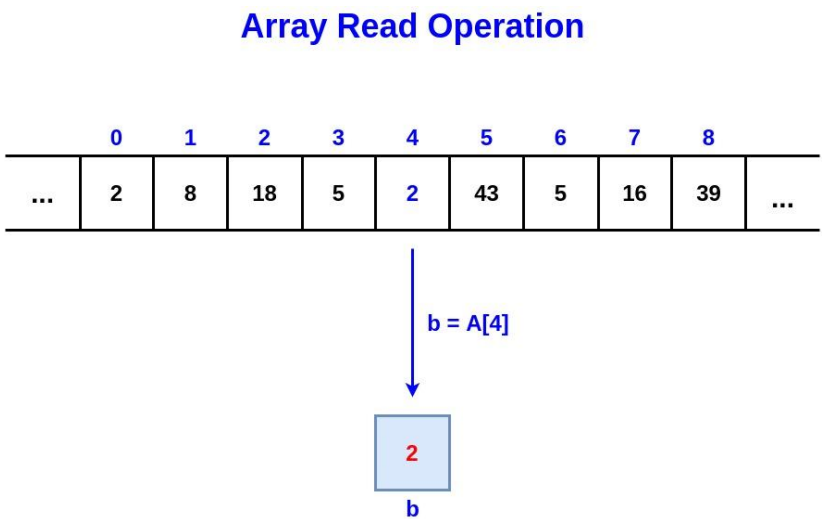
Array operations: “Insert”

- “Insertion” only means assigning a value to an array element (slot)
  - Is the same as update.
  - Uses indexing for the assignment.



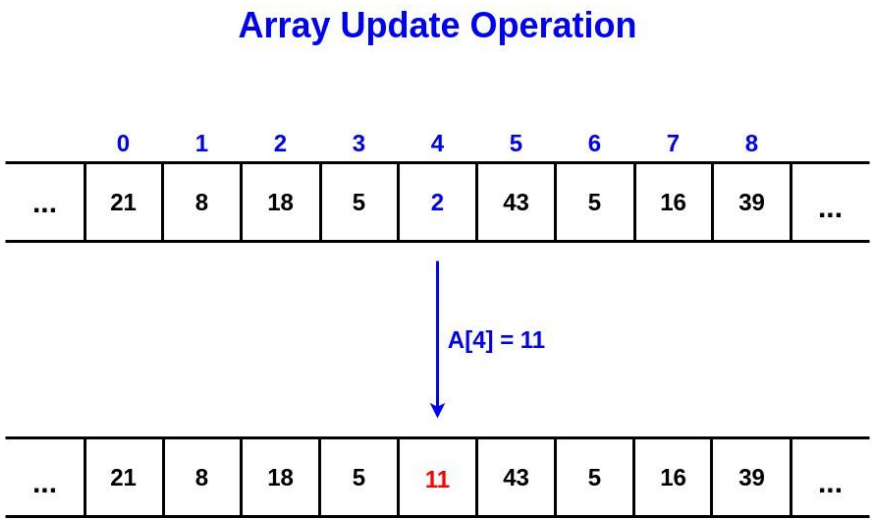
Array operations: Read

- A Read is the opposite of an “Insert”
  - Get the element value at a slot.
  - Uses indexing.



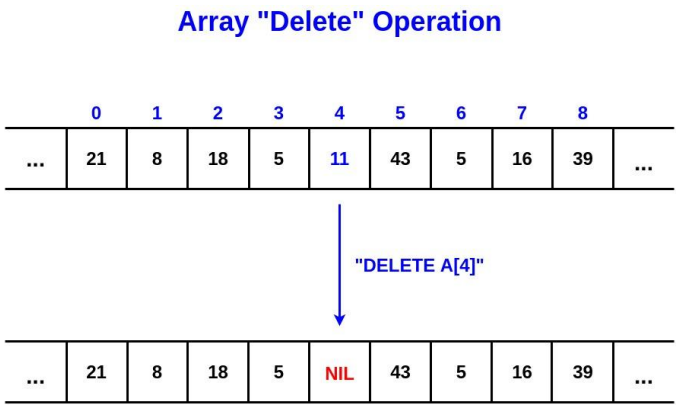
Array operations: Update

- An Update is similar to an “Insert”.
  - Changes the element value at a slot.
  - Uses indexing.



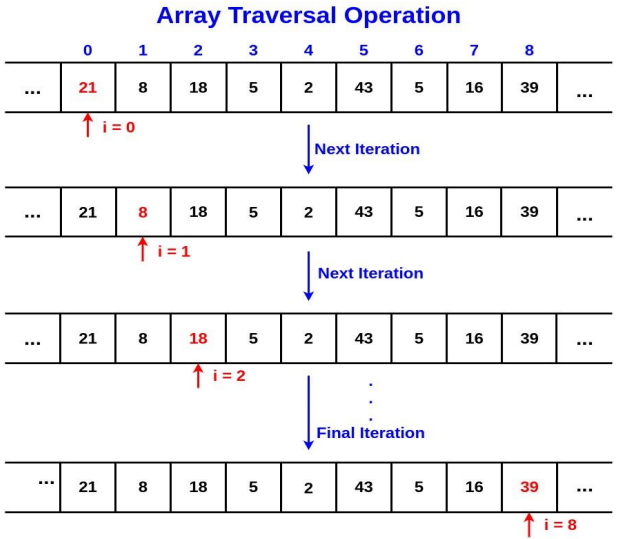
Array operations: “Delete”

- A “Delete” is logically just marking data in a slot INVALID, or NULL.
  - The slot storage remains where it is.
  - The slot can get a new element from an “Insert” later.
  - Uses Indexing.



Array operations: “Traversal”

- Since an array is random-access:
  - It can be traversed in any manner imaginable.
  - Depends only on the needs of the application.



Array operations: Complexity

- Analyzing time complexity
    - Create: Constant time (allocated as a block)
    - Insert: Constant time
    - Read: Constant time
    - Update: Constant time
    - Traversal: Proportional to N (size of the array)

Array operations: Complexity

- Analyzing space complexity:
  - Create: Proportional to N (one-time)
  - Insert: Constant space
  - Read: Constant space
  - Update: Constant space
  - Traversal: Constant space (in-place operation)

Summary

- We understood the concept of an array.
- We saw important properties of arrays.
- We saw the advantages array indexing offers.
- We explored the standard array operations.

Thank You