

# Static and Dynamic Arrays

Garrett Giles

July 20, 2022

## Outline

- Discussion and examples about Arrays
  - What is an Array?
  - When and where is a Array used?
  - Complexity
  - Static array usage examples
- Dynamic Array implementation details
- Code Implementation

## What is a static Array?

A static array is a fixed length container containing  $n$  elements indexable from the range  $[0, n-1]$ .

Q: What is meant by being 'indexable'?

A: This means that each slot/index in the array can be referenced with a number.

## When and where is a static Array used?

1. Storing and accesing sequential data
2. Temporarily storing object
3. Using by IO routines as buffers
4. Lookup table and inverse lookup tables
5. Can be used to return multiple values from a function
6. Used in dynamic programming to cache answers to subproblems

Table 1: Complexity

	<b>Static Array</b>	<b>Dynamic Array</b>
Access	$O(1)$	$O(1)$
Search	$O(n)$	$O(n)$
Insertion	N/A	$O(n)$
Appending	N/A	$O(1)$
Deletion	N/A	$O(n)$

## Static Array

$$A = \begin{bmatrix} 44 & 12 & -5 & 17 & 6 \\ 0 & 1 & 2 & 3 & 4 \end{bmatrix}$$

Elements in A are referenced by their index. There is no other way to access elements in an array. Array indexing is zero-based, meaning the first element is found in position zero.

## Operations on Dynamic Arrays

### Dynamic Array

The dynamic array can grow and shrink in size.

$A = [34, 4]$

$A.add(34) \Rightarrow A = [34, 4, 34]$

Q: How can we implement a dynamic array?

A: One way is to use a static array!

1. Create a static array with an initial capacity.
2. Add elements to the underlying static array, keeping track of the number of elements.
3. If adding another element will exceed the capacity, then create a new static array with twice the capacity and copy the original elements into it.

Suppose we create a dynamic array with an initial capacity of two and then begin adding elements to it.

$$\begin{bmatrix} \square & \square \end{bmatrix} \Rightarrow \begin{bmatrix} 7 & \square \end{bmatrix} \Rightarrow \begin{bmatrix} 7 & -9 & 3 & \square \end{bmatrix} \Rightarrow \begin{bmatrix} 7 & -9 & 3 & 12 \end{bmatrix}$$

Note the array will double when overflow occurs...

## Source code link

Implementation source code and tests can all be found at the following link:

[github.com/williamifset/data-structures](https://github.com/williamifset/data-structures)