Dynamic Arrays

Contents

- What are dynamic arrays?
- Creating Arrays
- Adding elements.
- Growing the array
- Removing elements
  - Ordered
  - Unordered

Dynamic Arrays

- Often you need to allocate memory for an array when you don’t know exactly how big it will need to be.

- A dynamic array is an array that grows and shrinks to roughly match the size of the data stored within it.

Dynamic Arrays

- The lifetime of a dynamic array looks something like this:
  - The array is created with some amount of space – typically more than is initially needed.
  - Elements are added to the array, filling up the allocated space.
  - If more elements are added to the array than there is space for a new, bigger array is allocated.
  - The memory is copied from the old array to the new one and the old one is deallocated.

Dynamic Arrays

- We will look at each of the common operations performed on a dynamic array to see you it works.

- We will look at:
  - Creating a dynamic array
  - Adding elements to the end
  - Adding elements in the middle
  - Removing elements from the end
  - Removing elements from the middle
  - Resizing the array

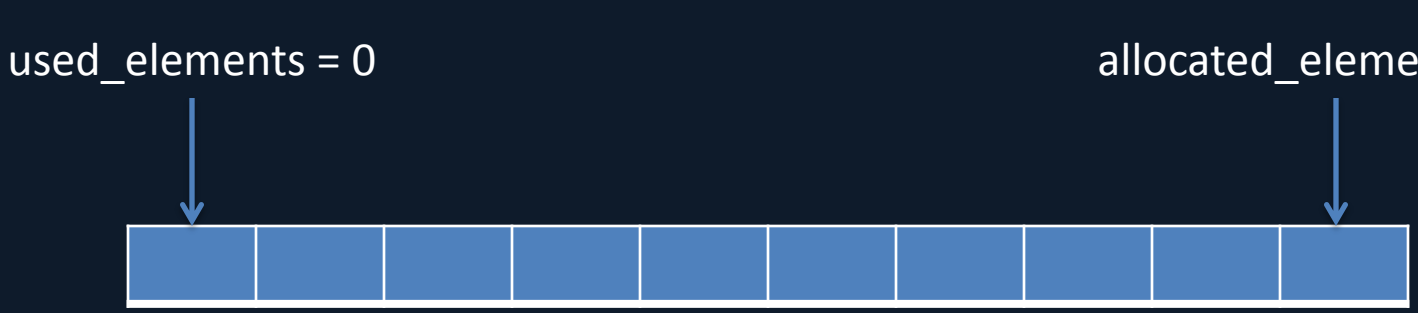
Creating a Dynamic Array

- Creating a dynamic array is very simple to begin with.

- We store 3 values:
  - The pointer to the array itself. The array is allocated on the heap.
  - A number storing how much memory we allocated
  - A number storing how many elements are currently in use.

Creating a Dynamic Array

- We create the dynamic array with a small amount of working data to start with.
- We keep track of how many elements are actually being used.



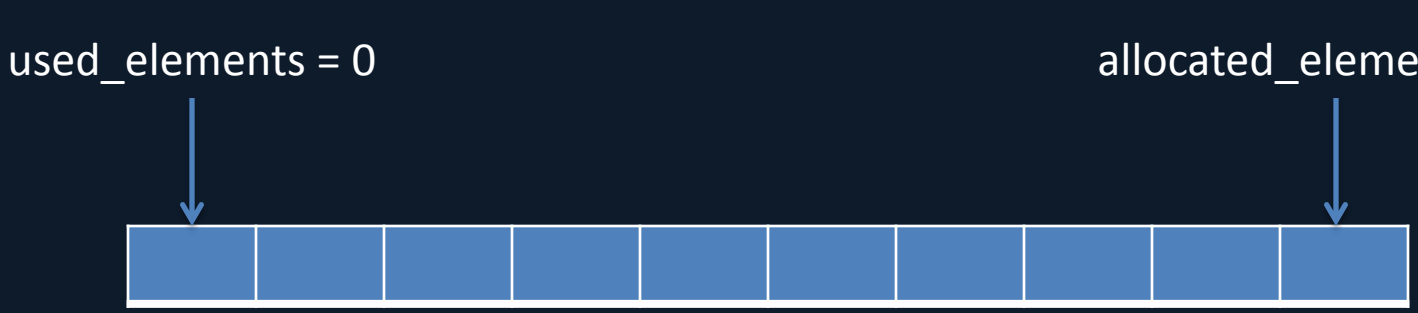
Creating a Dynamic Array

```
FUNCTION CreateArray(initial_size)
{
    DynamicArray array;
    array.data = new array[initial_size];
    array.allocated_elements = initial_size;
    array.used_elements = 0;

    RETURN array;
}
```

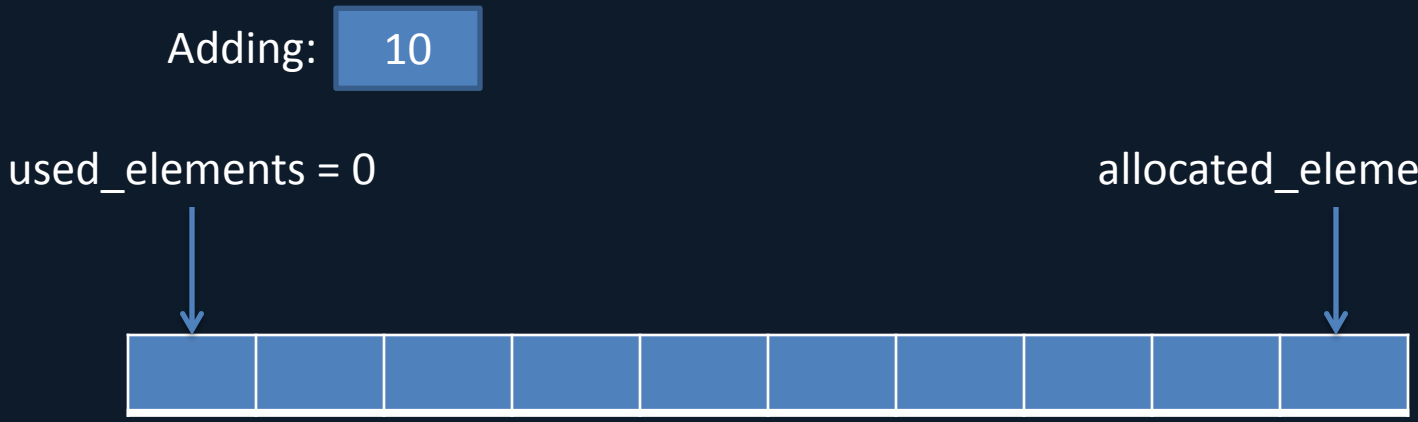
Adding to a Dynamic Array

- To add to a dynamic array, we just copy the new element into the first empty slot.



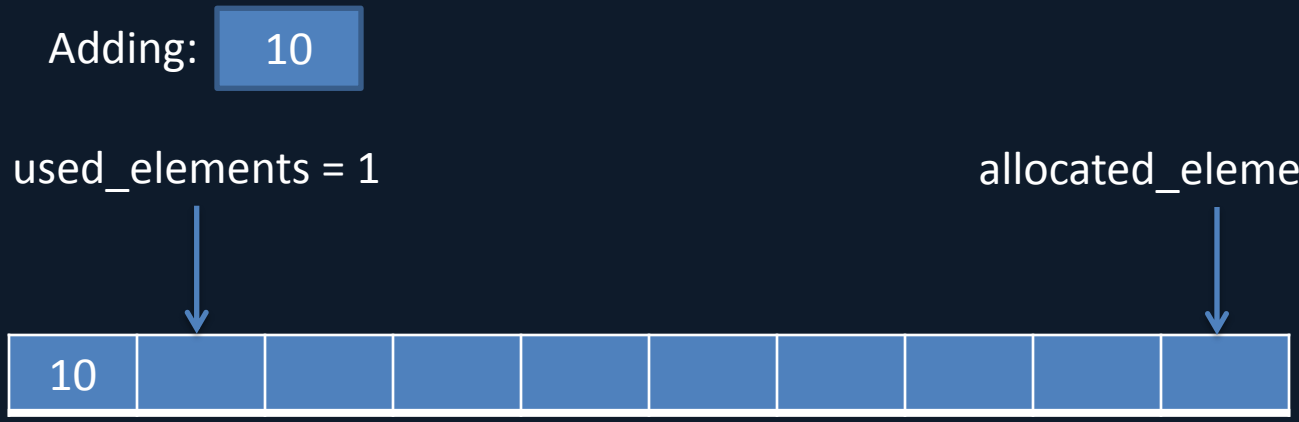
Adding to a Dynamic Array

- To add to a dynamic array, we just copy the new element into the first empty slot.



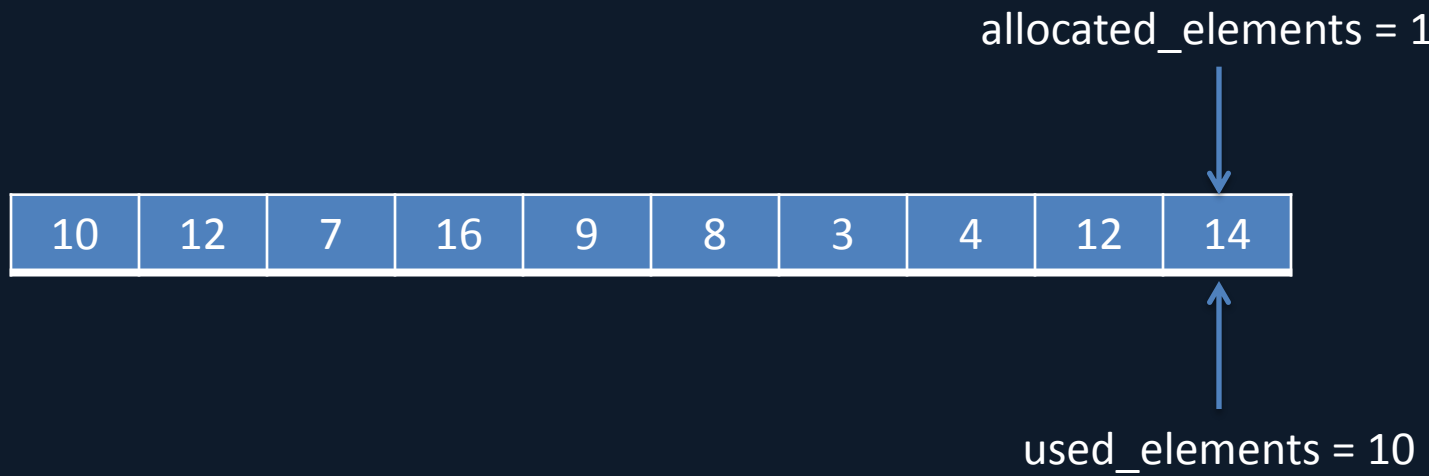
Adding to a Dynamic Array

- To add to a dynamic array, we just copy the new element into the first empty slot.



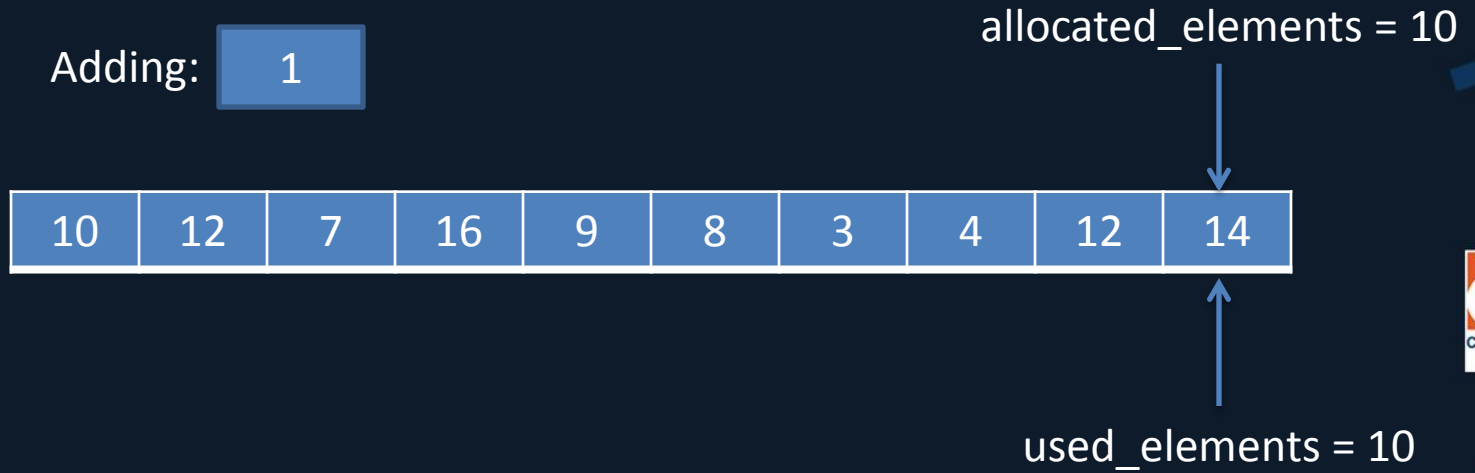
Adding to a Dynamic Array

- The problem comes when we run out of space.



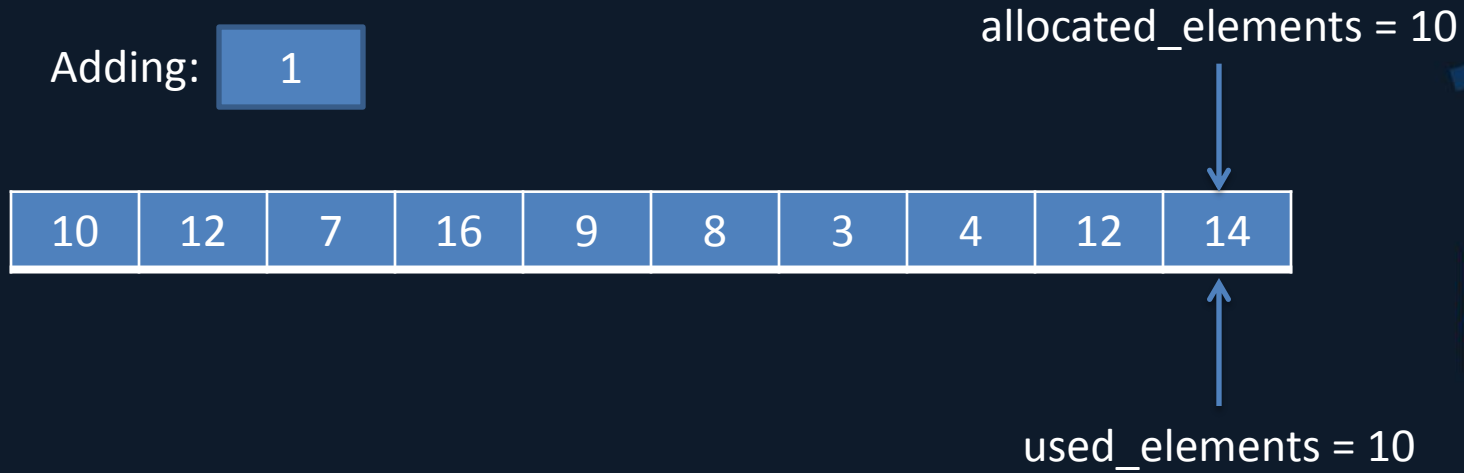
Adding to a Dynamic Array

- The problem comes when we run out of space.



Adding to a Dynamic Array

- We allocate a new array that is bigger, copy the old array into it and delete it.

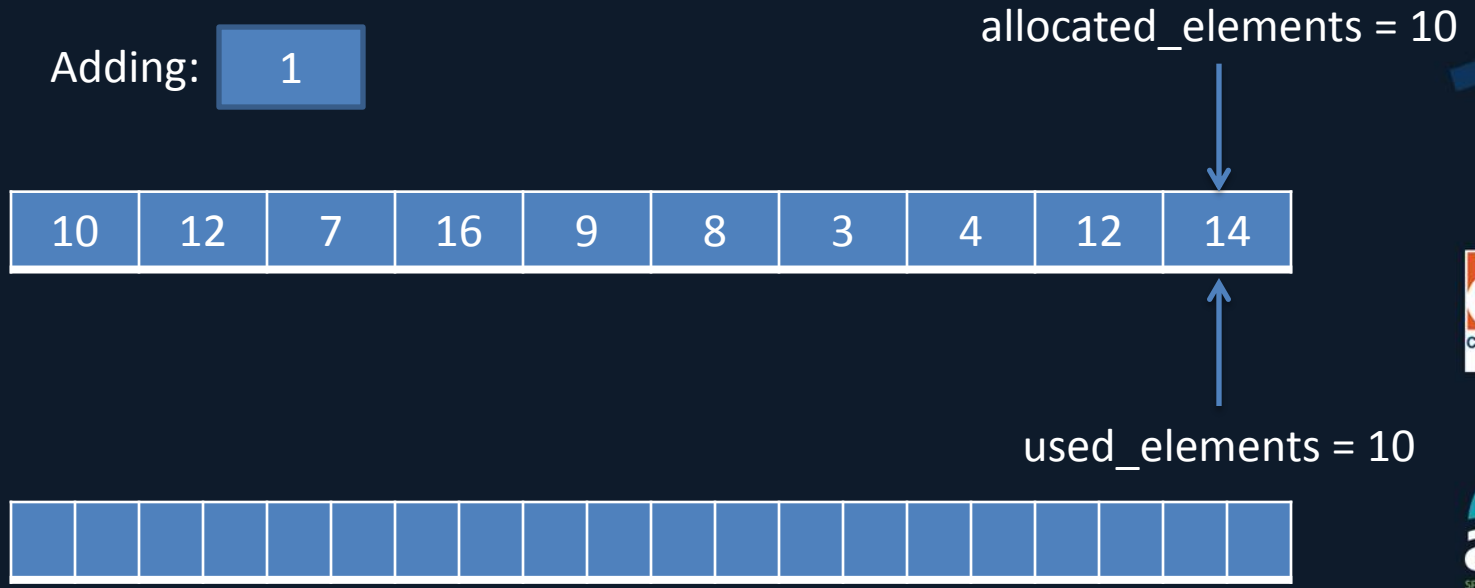


Canberra Institute of Technology

aie
SPECIALIST EDUCATORS IN GAMES, ANIMATION & FILM VFX

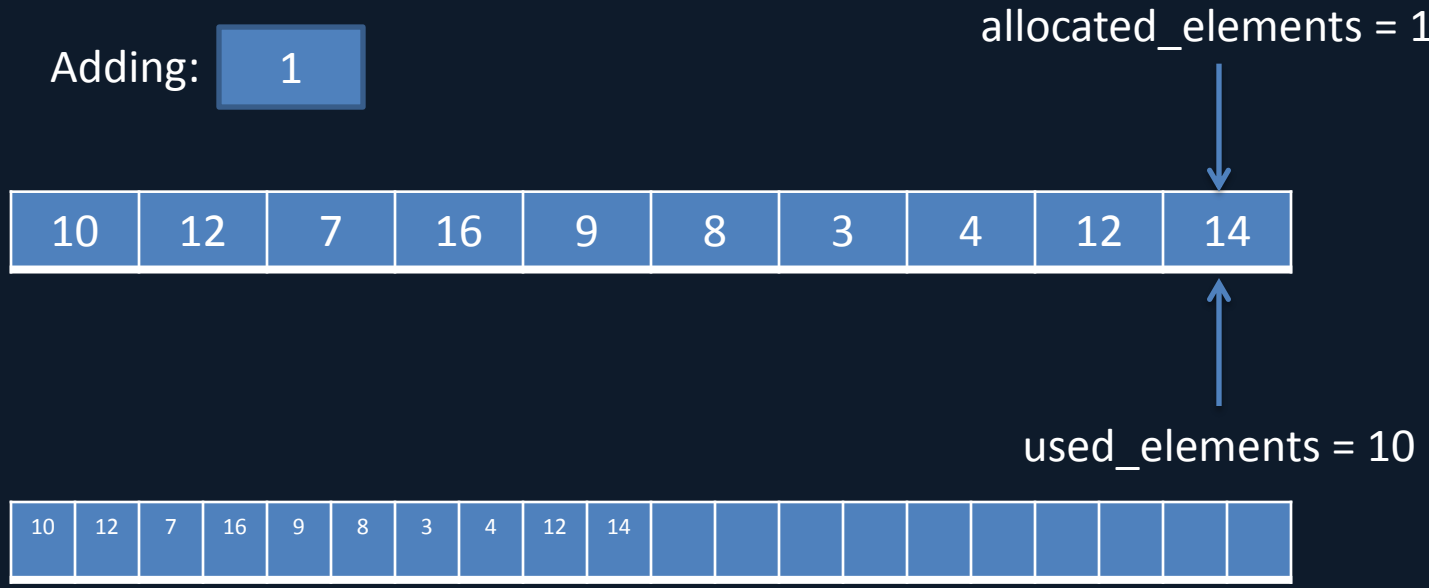
Adding to a Dynamic Array

- We allocate a new array that is bigger, copy the old array into it and delete it.



Adding to a Dynamic Array

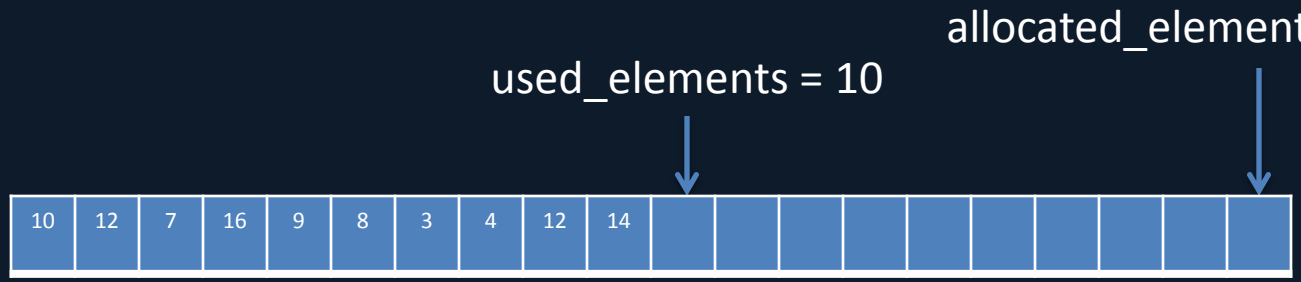
- We allocate a new array that is bigger, copy the old array into it and delete it.



Adding to a Dynamic Array

- We allocate a new array that is bigger, copy the old array into it and delete it.

Adding: 1



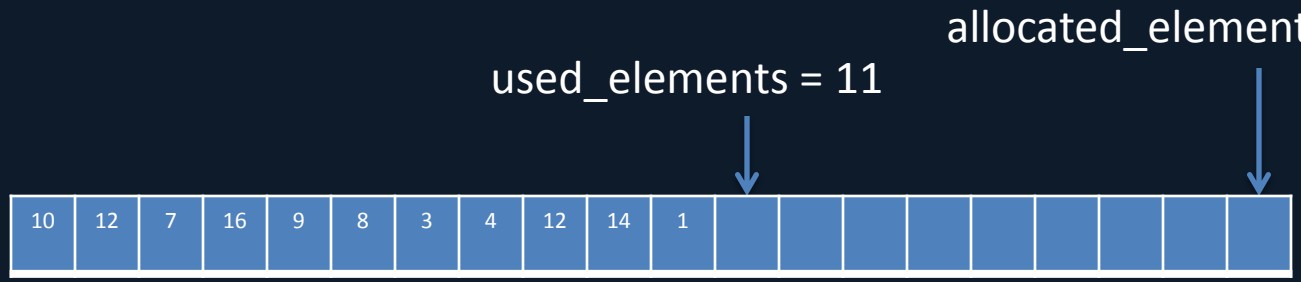
Canberra Institute of Technology

aie
SPECIALIST EDUCATORS IN GAMES, ANIMATION & FILM VFX

Adding to a Dynamic Array

- We allocate a new array that is bigger, copy the old array into it and delete it.

Adding: 1



Canberra Institute of Technology

aie
SPECIALIST EDUCATORS IN GAMES, ANIMATION & FILM VFX

Adding to a Dynamic Array

```
FUNCTION AddToArrayEnd(array, new_element)
{
    IF array.allocated_elements == array.used_elements THEN
        new_data = new array[array.allocated_elements * 2];
        copy(new_data, array.data, array.allocated_elements);
        delete array.data
        array.data = new_data;
        array.allocated_elements *= 2
    END IF

    array.data[array.used_elements] = new_element;
    array.used_elements += 1;
}
```

Adding to a Dynamic Array

- This covers adding to the end of the array.
- Adding to the middle is similar.
  - The same size check must be done first.
- However, instead we must first shift all elements after where you want to add forward by one slot.

Expanding the Array

- In the example given, we double the size of the array every time we run out of space.
- This could lead to large amounts of memory wasted.
- You might want to only expand the array by 50% every time it needs to grow
  - Or 25%
  - Or 10%
  - Or only by a single element
- The less you expand the array by, the closer it will match the exact amount of memory used. However, the smaller the expansion, the more often you will need to allocate, copy and deallocate the array.

Removing From a Dynamic Array

- Removing from the end of a dynamic array is very easy.
- Just subtract 1 from the used_elements variable.
- We don’t need to modify the existing data. It becomes the same as the junk data that the allocated array starts as.

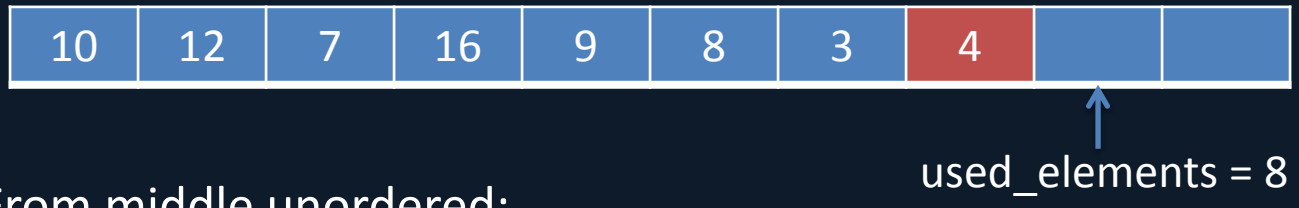
Removing From a Dynamic Array

- Removing from the middle is a bit harder and provides us with a choice
  - If we don't care about the order of elements in the array we can just copy the last element over the top of the element we want to remove
  - If we do care about order, we have to move all elements after the end of the array back by one element.

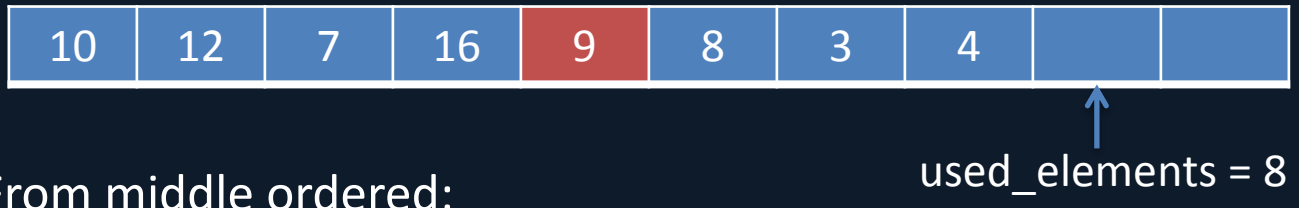
- After doing either of these, we decrement used_elements like before.

Removing from a dynamic array

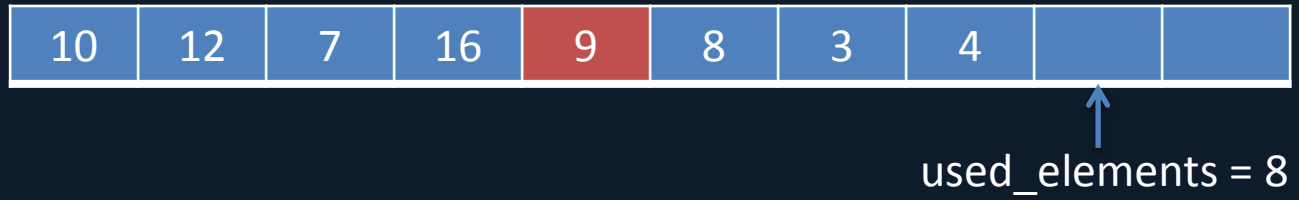
- From end:



- From middle unordered:



- From middle ordered:

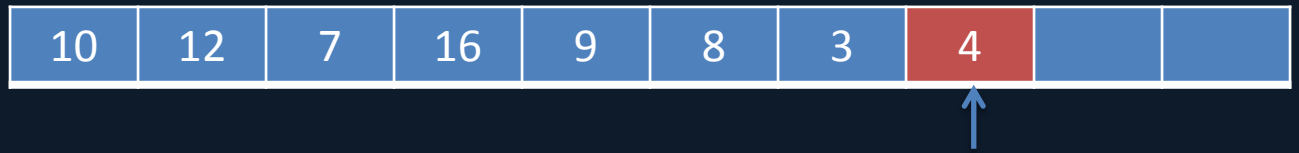


Canberra Institute of Technology

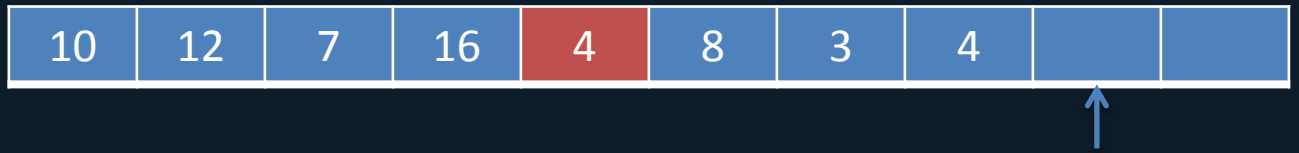
aie
SPECIALIST EDUCATORS IN GAMES, ANIMATION & FILM VFX

Removing from a dynamic array

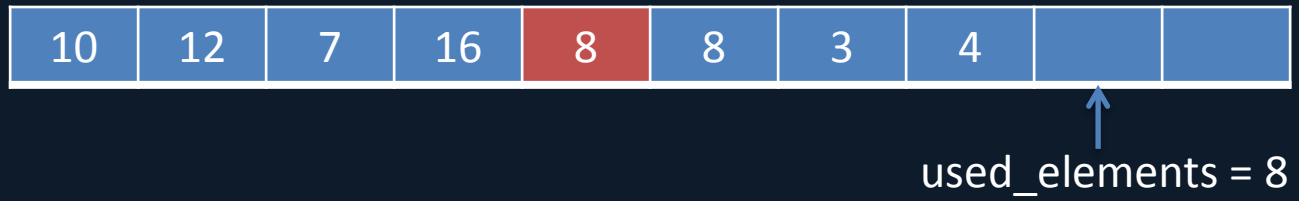
- From end:



- From middle unordered:



- From middle ordered:

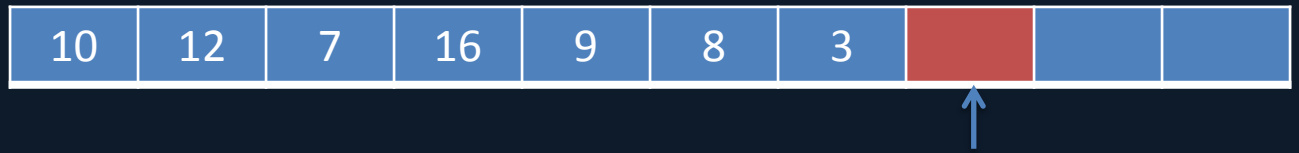


Canberra Institute of Technology

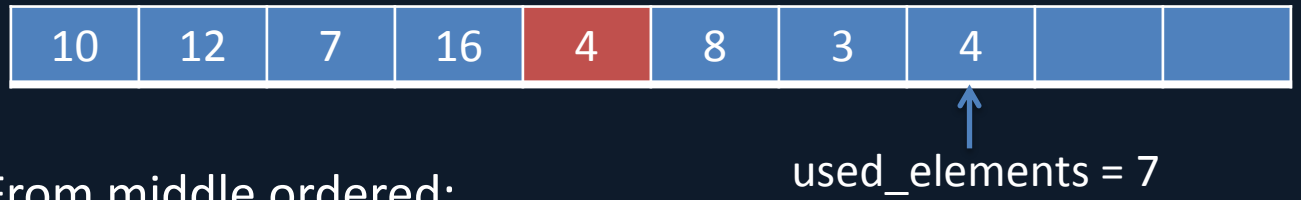
aie
SPECIALIST EDUCATORS IN GAMES, ANIMATION & FILM VFX

Removing from a dynamic array

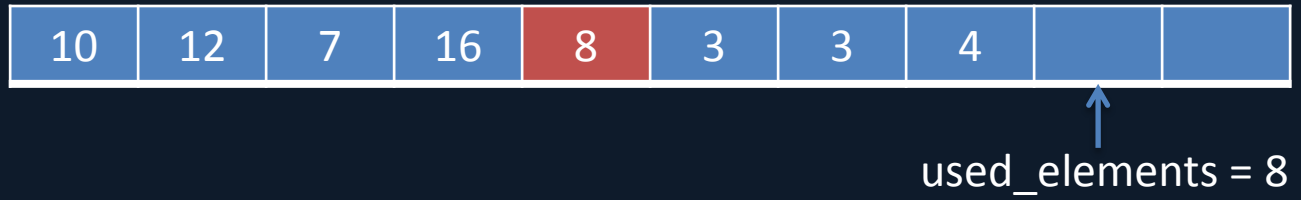
- From end:



- From middle unordered:



- From middle ordered:

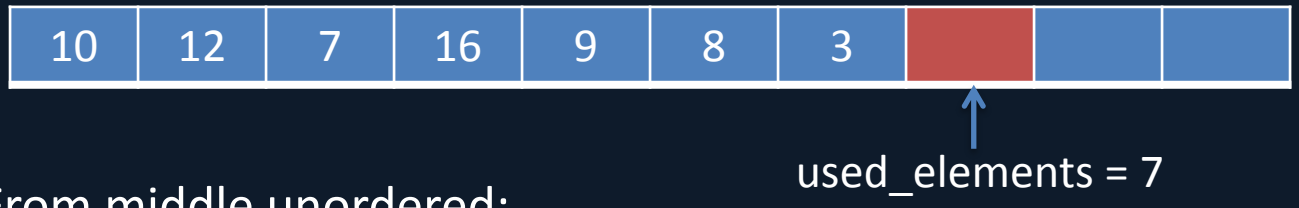


Canberra Institute of Technology

aie
SPECIALIST EDUCATORS IN GAMES, ANIMATION & FILM VFX

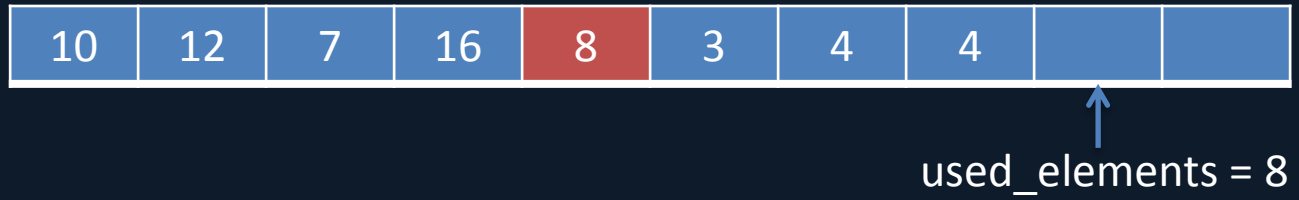
Removing from a dynamic array

- From end:



- From middle unordered:

- From middle ordered:

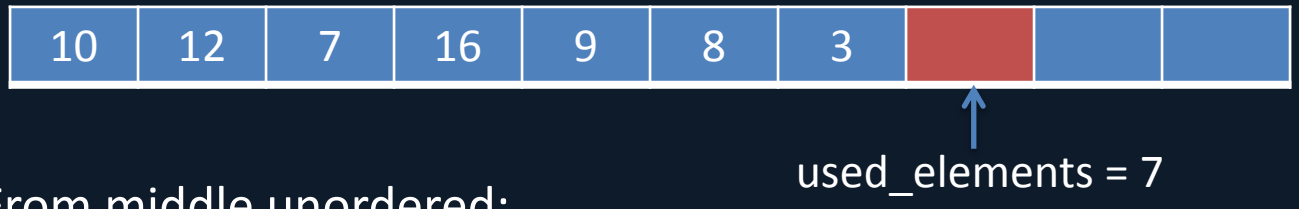


Canberra Institute of Technology

aie
SPECIALIST EDUCATORS IN GAMES, ANIMATION & FILM VFX

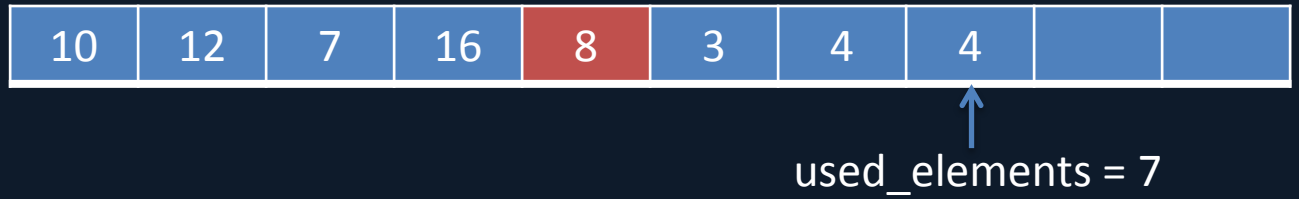
Removing from a dynamic array

- From end:



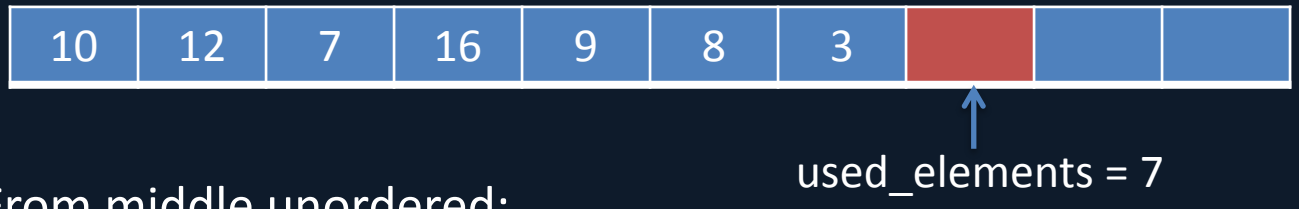
- From middle unordered:

- From middle ordered:



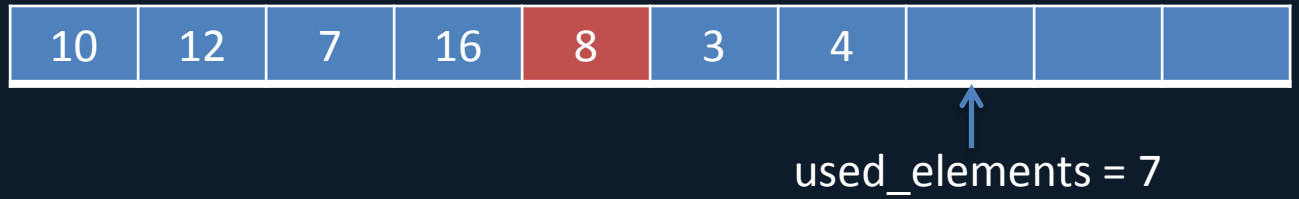
Removing from a dynamic array

- From end:



- From middle unordered:

- From middle ordered:



Canberra Institute of Technology

aie
SPECIALIST EDUCATORS IN GAMES, ANIMATION & FILM VFX

Summary

- Dynamic arrays are a commonly used tool in programming.
- They allow you to store data in situations where it is hard or even impossible to know how much data will be needed prior to storing it.
- They are simple to implement.
- You could add more features than were described here:
  - Functions that change the allocated size
  - Functions that add and remove more than one element at a time.
  - Functions that clear all elements in the array

References

- Sedgewick, R, and Wayne, K “*Algorithms*”, 4th Ed, Chp 1, Addison-Wesley (2011)

Canberra Institute of Technology

aie
SPECIALIST EDUCATORS IN GAMES, ANIMATION & FILM VFX