19CSE100 Problem Solving and Algorithmic Thinking

Data Organization
Lists, Arrays

Organizing data

- Humans hate disorder
- We try and organize our lives, our cities, our homes
- There are many ways of organizing things, and we naturally organize different things in different ways
- The organization we use depends on the things we are trying to organize and often more importantly, what we actually wish to do with them once they are organized

Organizing data

- Addresses are organized into postal areas so that it is easier for letters to be delivered.

- A bookshop organizes its novels on the shelves usually alphabetically by author

- The way we organize the things we must manipulate to achieve a task can make a massive difference to how quickly, easily or successfully the task can be done

Is there a better way to store and access related items?

- *How should I store the details of all the student names of my class?*

    - *If 60 students in a class can I have 60 different variable names?*
    - E.g. name1,name2, name3 ……., name60.

    ✓ *Is it organized for efficient access?*

    ✓ *Is it easy to be remembered?*

    ✓ *What happens if number of students increase from 60 to 80?*

Variables you've seen so far store single unit of data

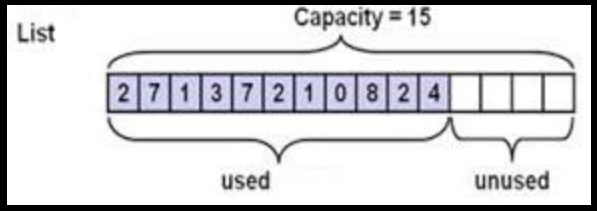
What if your application demands a collection of data to be processed?

Collection of data usually comprises related information

Good idea to store them together – not only logical but also efficient!!

Lists

- A list is a sequence of items that are arranged in a particular order.
- Eg. list of your all time favorite records
    1. Bohemisn Rhspdody – Queen
    2. Stairway to Heaven – Led Zeppelin
    3. Your Song – Elton John
    4. Who made who – AC/DC
    5. Sweet Child of Mine – Guns ‘n Roses
- Any item on the list can be identified by its position in the list.
- We denote the first item in the list as list [first_location] and the second item in the list as list[second_location].



Lists – Storage organization

- Data is stored at some location and each location has an address (Figure: Storage Organization)

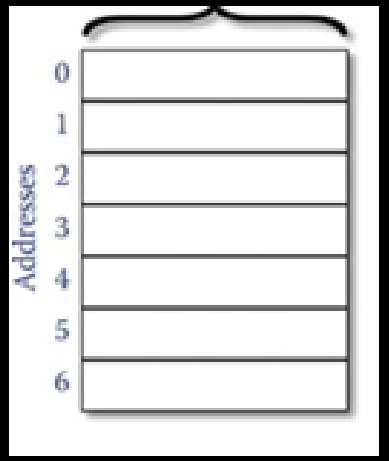


Figure : Storage Organization

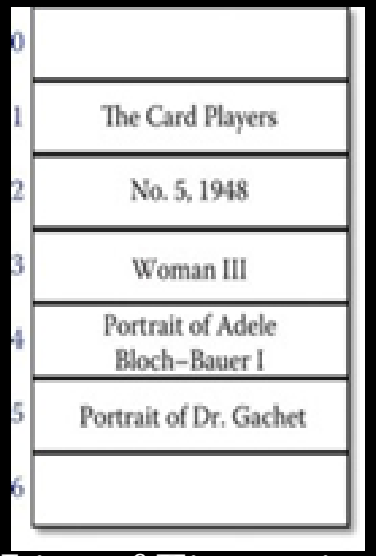


Figure: List of Five painting records

Each piece of data is located at an address (Figure: List of Five painting records).

Lists - Indices

- Addresses in lists are numbers (indices) as illustrated in Figure: List indices

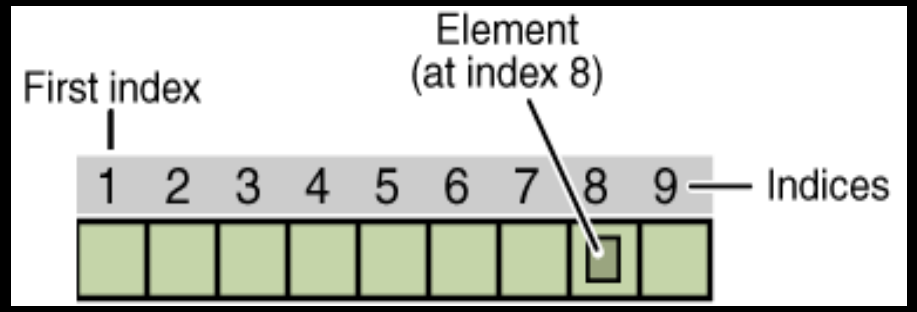


Figure: List indices

Adding Elements to list

- Lists can have new entries added anywhere, including in the middle, and their length can increase as needed.

Original List

1. *Bohemisn Rhspdody – Queen*
2. *Stairway to Heaven – Led Zeppelin*
3. *Your Song – Elton John*
4. *Who made who – AC/DC*
5. *Sweet Child of Mine – Guns'n Roses*

List after adding a new element

1. *Bohemisn Rhspdody – Queen*
2. *Every Breath you take - Police*
3. *Stairway to Heaven – Led Zeppelin*
4. *Your Song – Elton John*
5. *Who made who – AC/DC*
6. *Sweet Child of Mine – Guns 'n Roses*
7. *The Ghost of Tom Jord – Bruce Springsteen*

Removing Elements from List

- Entries can be removed from lists

Original List

1. *Bohemisn Rhspdody – Queen*
2. *Every Breath you take - Police*
3. *Stairway to Heaven – Led Zeppelin*
4. *Your Song – Elton John*
5. *Who made who – AC/DC*
6. *Sweet Child of Mine – Guns ‘n Roses*
7. *The Ghost of Tom Jord – Bruce Springsteen*

List after deletion of two elements

1. *Bohemisn Rhspdody – Queen*
2. *Stairway to Heaven – Led Zeppelin*
3. *Your Song – Elton John*
4. *Who made who – AC/DC*
5. *Sweet Child of Mine – Guns‘n Roses*

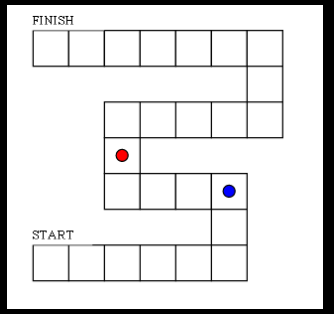
Real Time Applications of lists

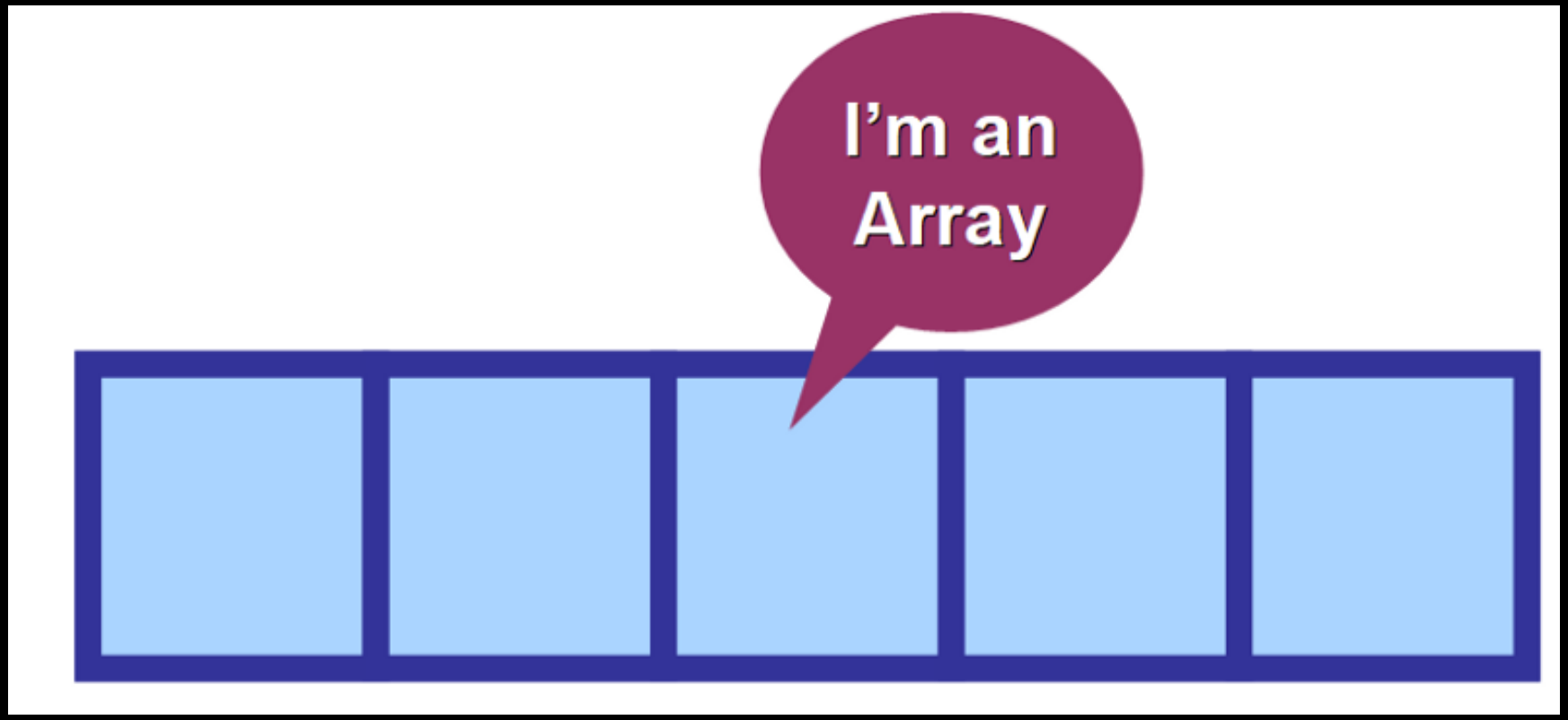
We use lists all the time…

✓Whenever I go shopping

✓The "To-Do" list is one of the most common ways that people use to organize their work or studies.

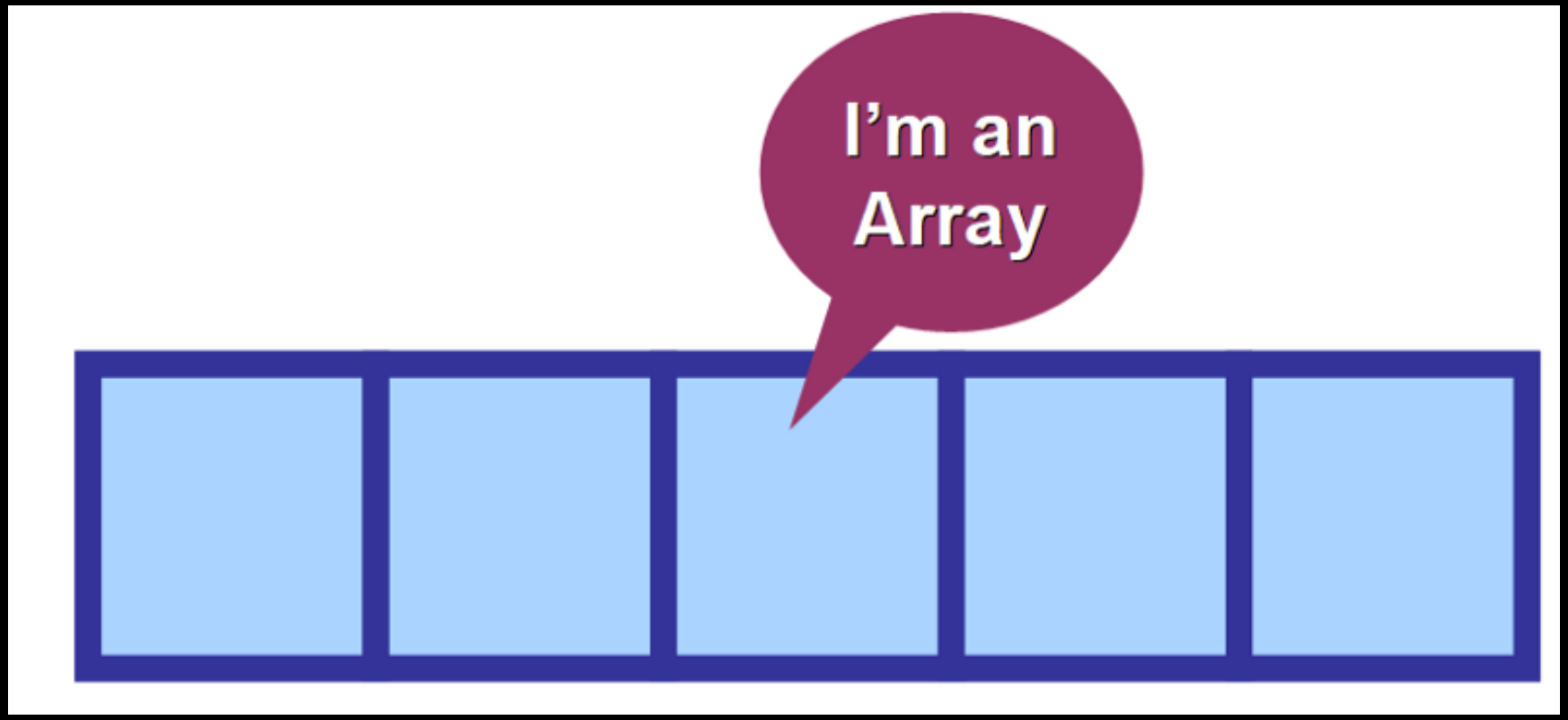
✓Many simple children's board games are based on list-like structures.





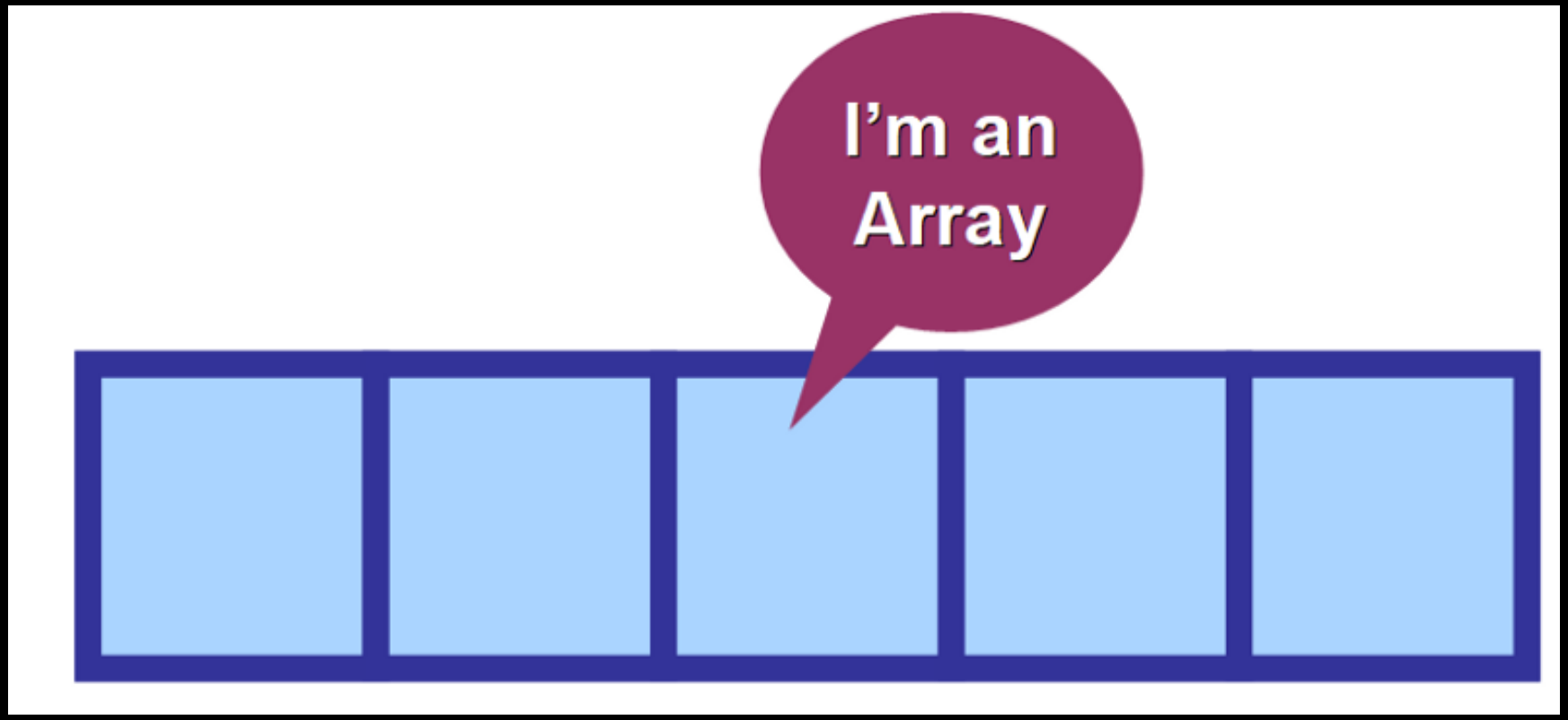
Arrays store list of related information in contiguous memory locations

https://www.freecodecamp.org/news/data-structures-101-arrays-a-visual-introduction-for-beginners-7f013bcc355a/



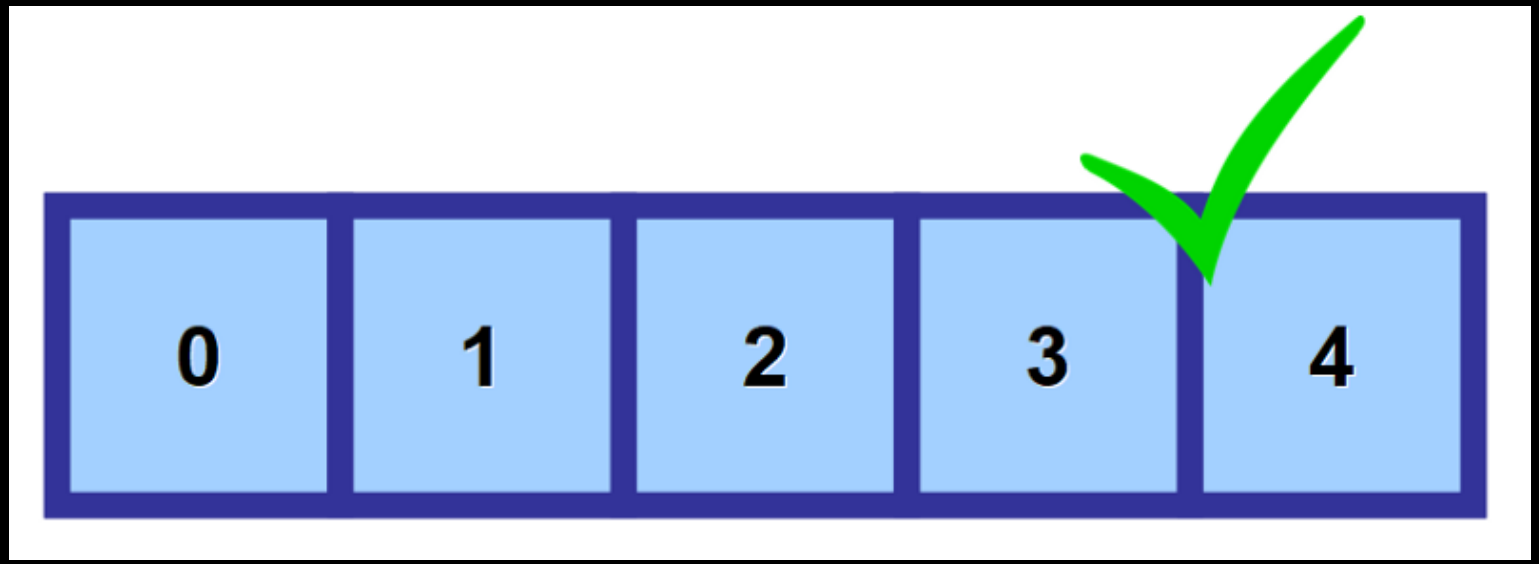
Each memory unit holds a unit of data – called elements of array. These elements are next to each other in memory location.

https://www.freecodecamp.org/news/data-structures-101-arrays-a-visual-introduction-for-beginners-7f013bcc355a/



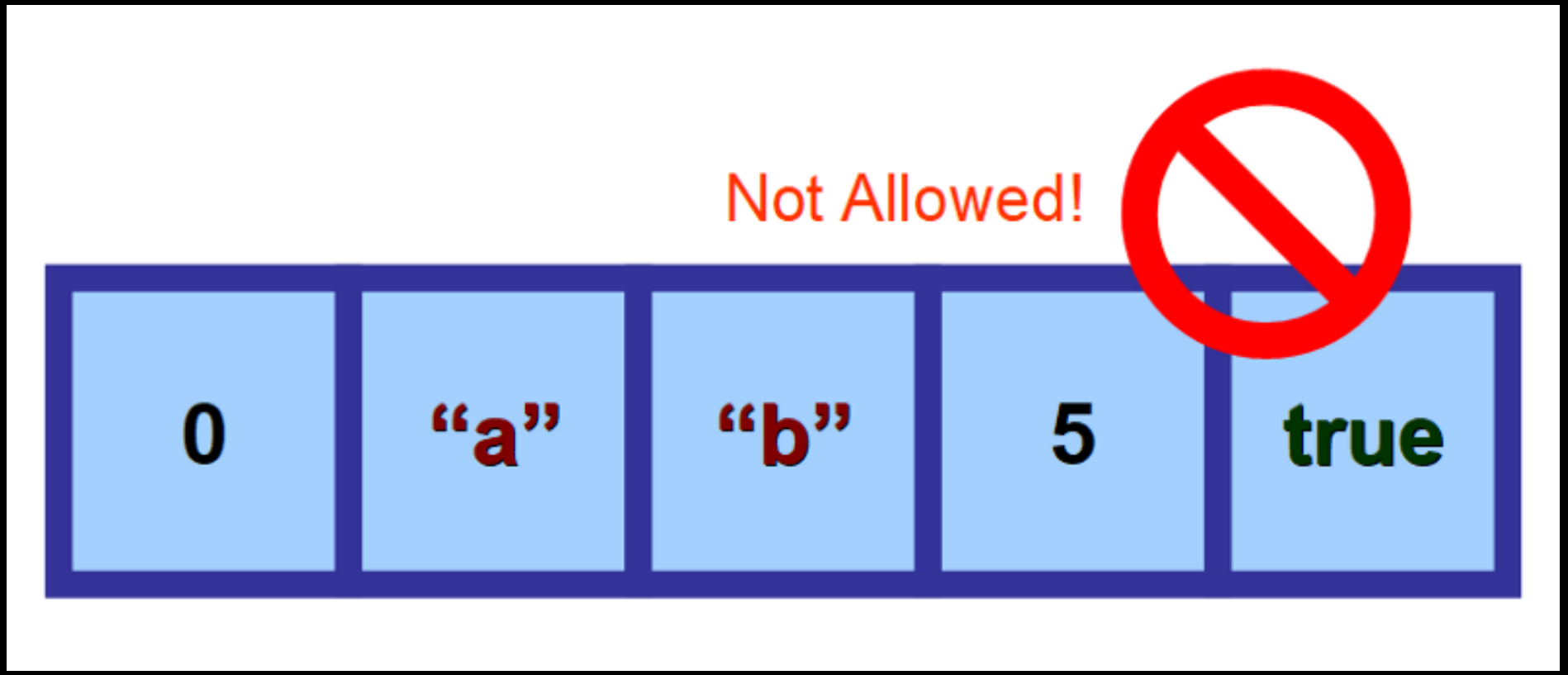
Since you often need more than one of them, the fetching process is extremely optimized because your computer knows where they are!

https://www.freecodecamp.org/news/data-structures-101-arrays-a-visual-introduction-for-beginners-7f013bcc355a/



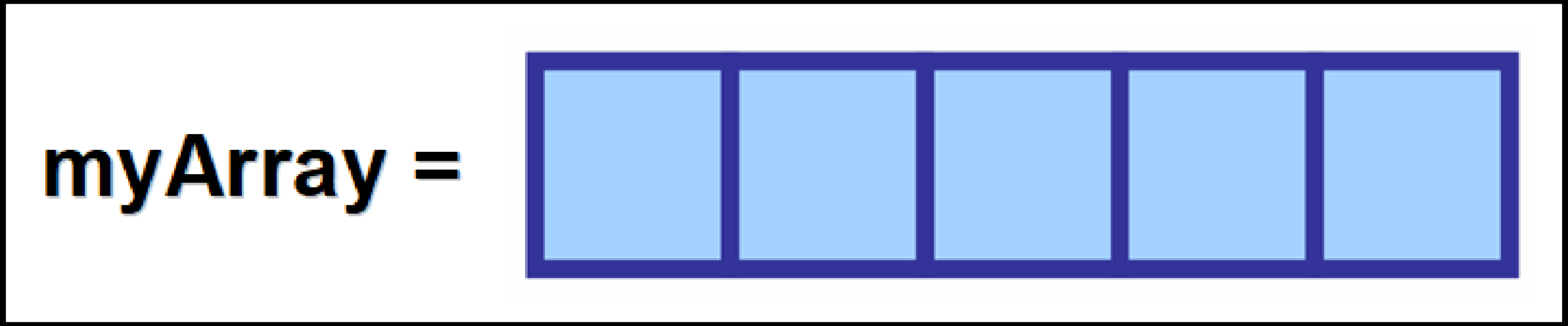
Arrays can store any datatypes (Integers, Reals, Booleans and Strings). But once you decide the data type, all the array elements must be of that same type. You can't mix different types of data!!

https://www.freecodecamp.org/news/data-structures-101-arrays-a-visual-introduction-for-beginners-7f013bcc355a/



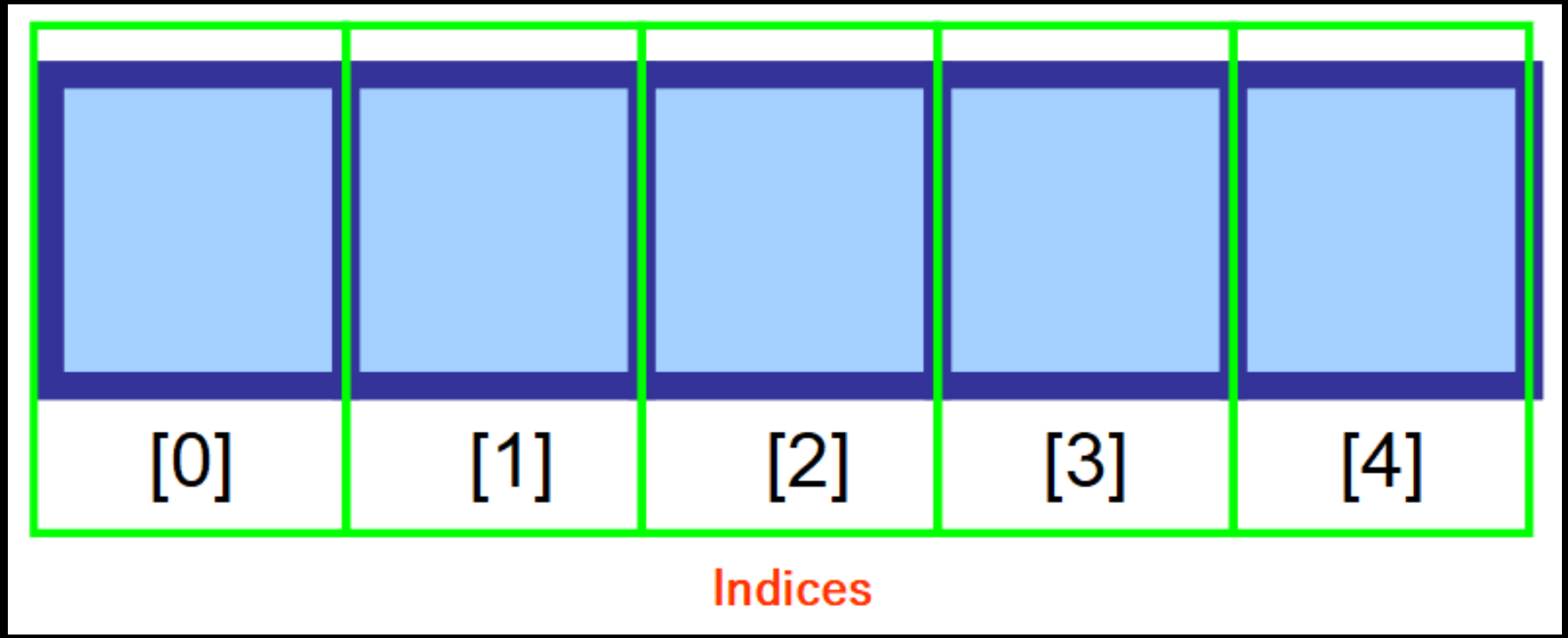
But lists can do that! Not every language has lists. Python has!!

https://www.freecodecamp.org/news/data-structures-101-arrays-a-visual-introduction-for-beginners-7f013bcc355a/



Now the array variable can hold multiple units of data. Typically the name also includes a square bracket that contains the size of the array. For example myArray[5]

https://www.freecodecamp.org/news/data-structures-101-arrays-a-visual-introduction-for-beginners-7f013bcc355a/



You use index (plural indices) to access a value in an array. This is a number that refers to the relative location where the value is stored or to be stored. Note zero-based-numbering

https://www.freecodecamp.org/news/data-structures-101-arrays-a-visual-introduction-for-beginners-7f013bcc355a/

Boundaries of an Array

- Arrays are by default zero-based, which means the enumeration of the elements starts from 0.
- The index of the first element is 0, and that of the second element is 1, and so on.
- In an array of N elements, the last element has the index N-1.

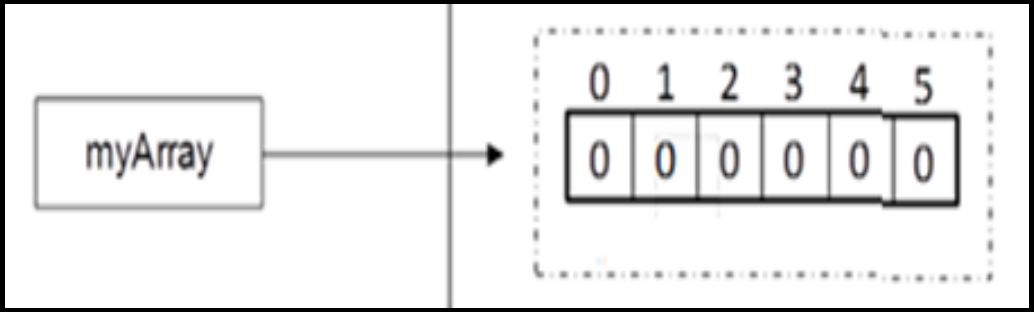


Figure: Boundary of myArray is 5

Array element access

- We access the array elements directly using their indices
- Each element can be accessed through the name of the array and the element's index (consecutive number) placed in the brackets
- For Eg. numberArray[2] produces the 3rd element, 6

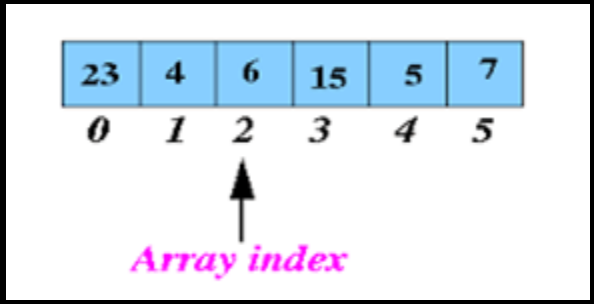
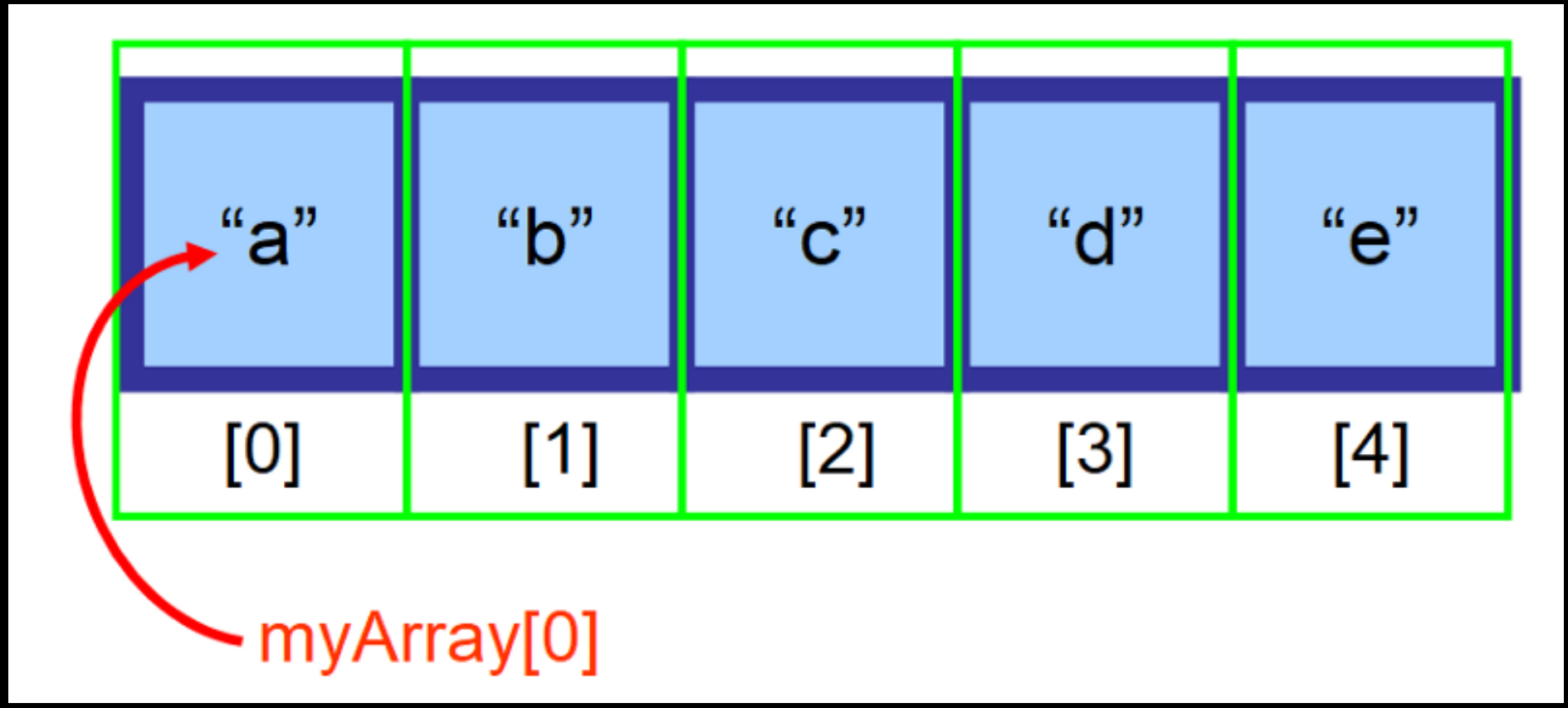


Figure: Elements of numberArray

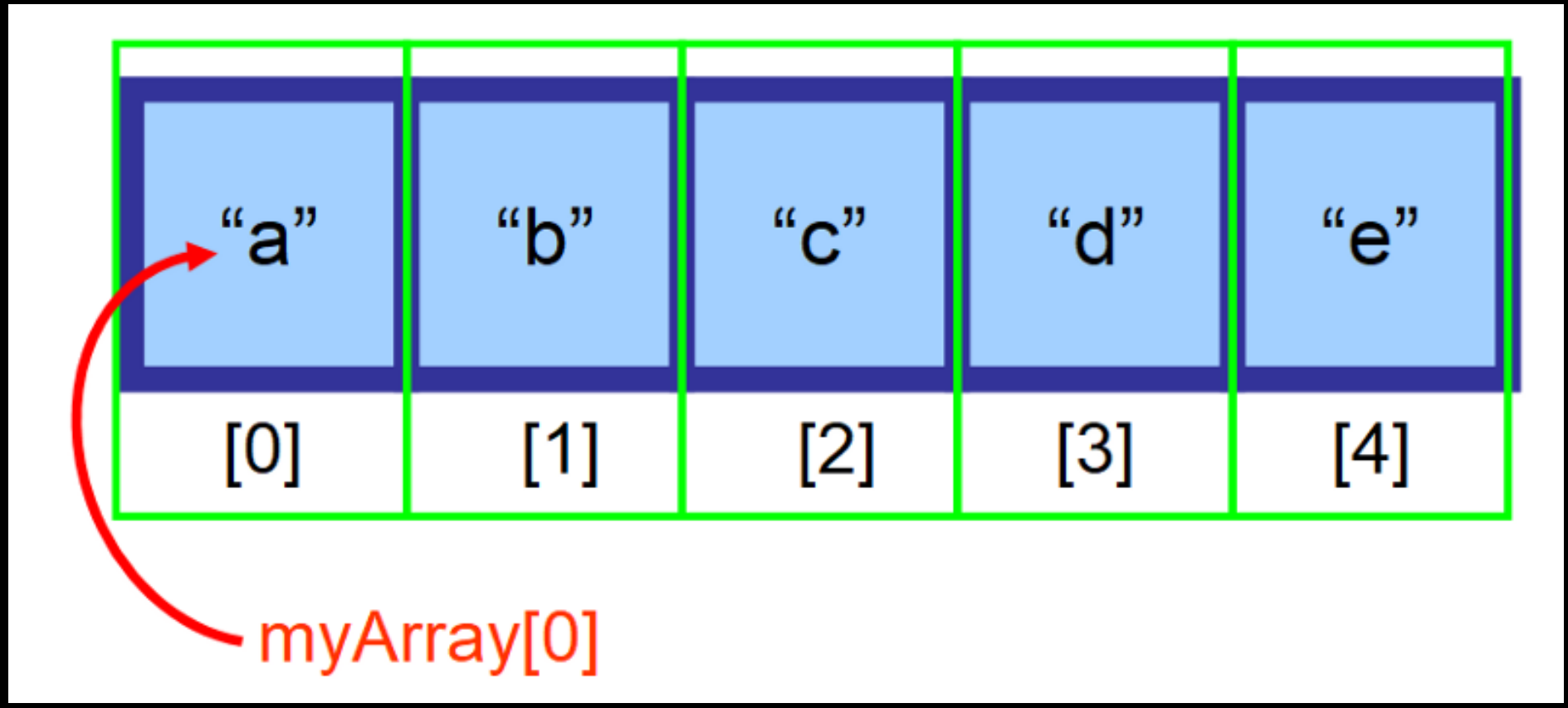
Array element access

- As long as we keep track of the location of the array, we will be able to access every item in the array.
- The location of the array is known as the base address or the anchor, and is the memory address of the first item in the array.
- From a computer's perspective, the name of an array corresponds to the anchor of the array.
  - If the name of the array is known, then the anchor is known and all of the array items can be accessed by their index.



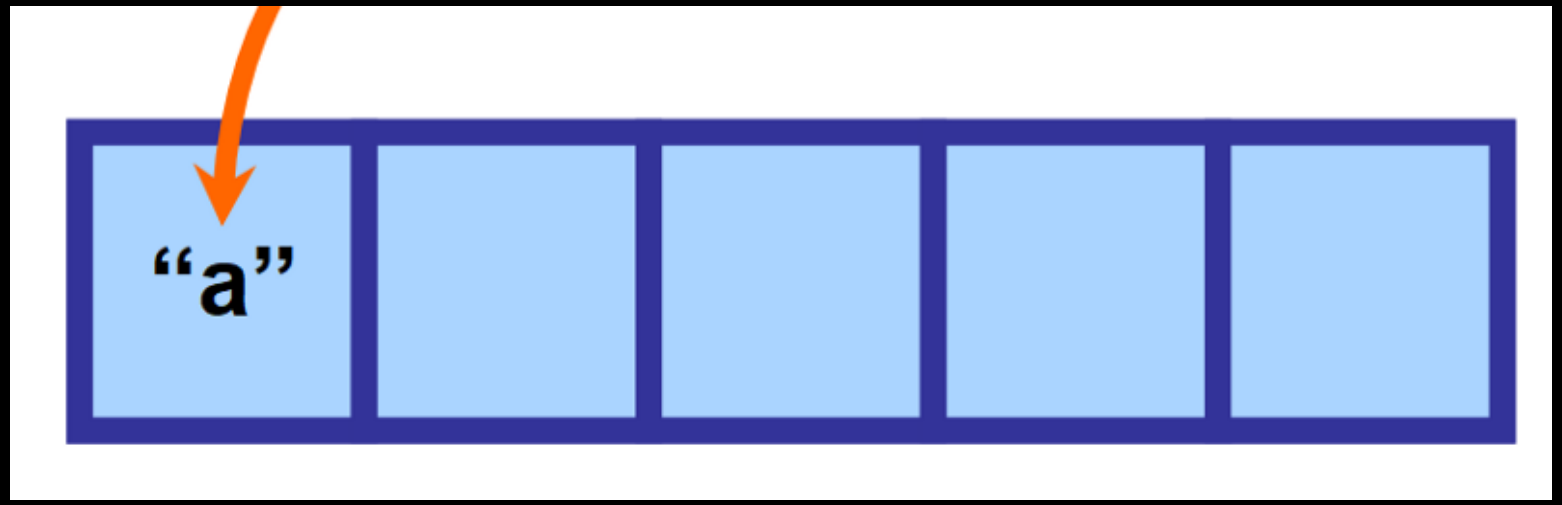
The general syntax to refer/access an element is given above.

https://www.freecodecamp.org/news/data-structures-101-arrays-a-visual-introduction-for-beginners-7f013bcc355a/



The general syntax to refer/access an element is given above.

https://www.freecodecamp.org/news/data-structures-101-arrays-a-visual-introduction-for-beginners-7f013bcc355a/



When you define the size of an array, all of that space is reserved. If you do not fill the array, that space is kept reserved and empty until you do.

https://www.freecodecamp.org/news/data-structures-101-arrays-a-visual-introduction-for-beginners-7f013bcc355a/

Array Properties

- *Arrays have three important properties.*
  - Arrays represent a group of related data.
    (e.g. -- temperature for the last five days, or stock prices for the last 30 days.)
  - All data within a single array must share the same data type.
  - The size of an array is fixed once it is created.

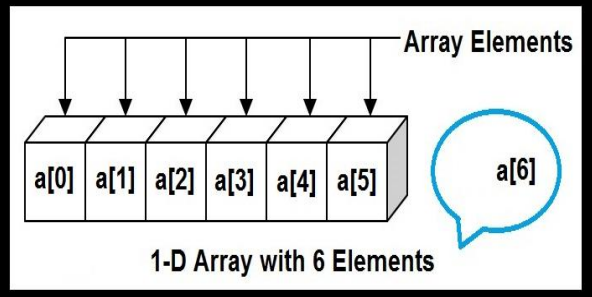
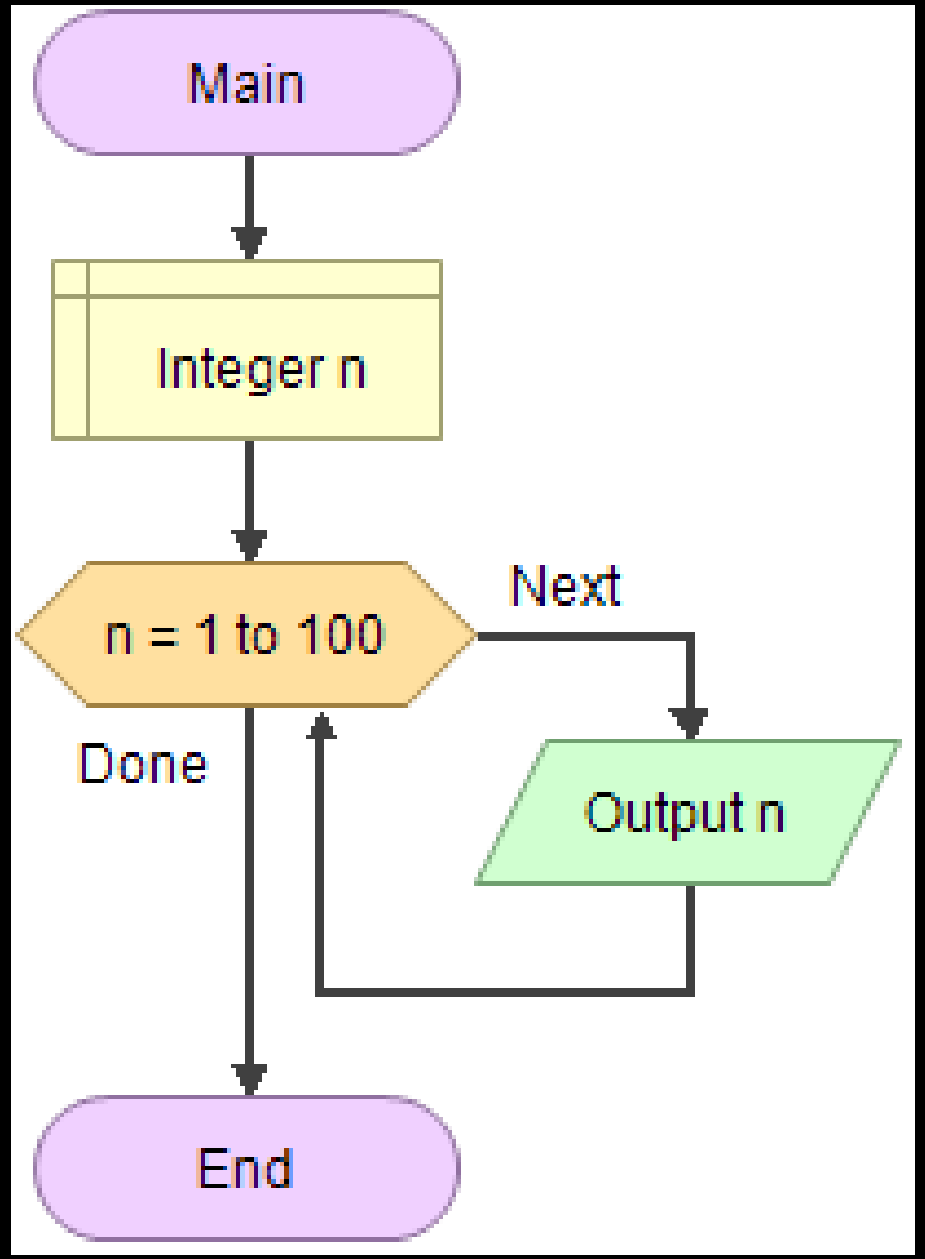


Figure: A 1-D array



This is why arrays are recommended when you know beforehand how many elements you are going to store. This is also the reason why often arrays and for loops go hand-in-hand.

1. *Given the array scores,*

scores

79	87	94	82	67	98	87	81	74	91

- *Can you find*

✓ *How many elements are there?*

✓ *What is the fifth element in the array?*

✓ *What is the result of mean = (scores[0]+scores[1])/2*

2. *Workout the average of Quiz1and student0 with the following data?*

	Quiz1	Quiz2
Student0	95	85
Student1	89	80

Answers

How many elements are there? 10

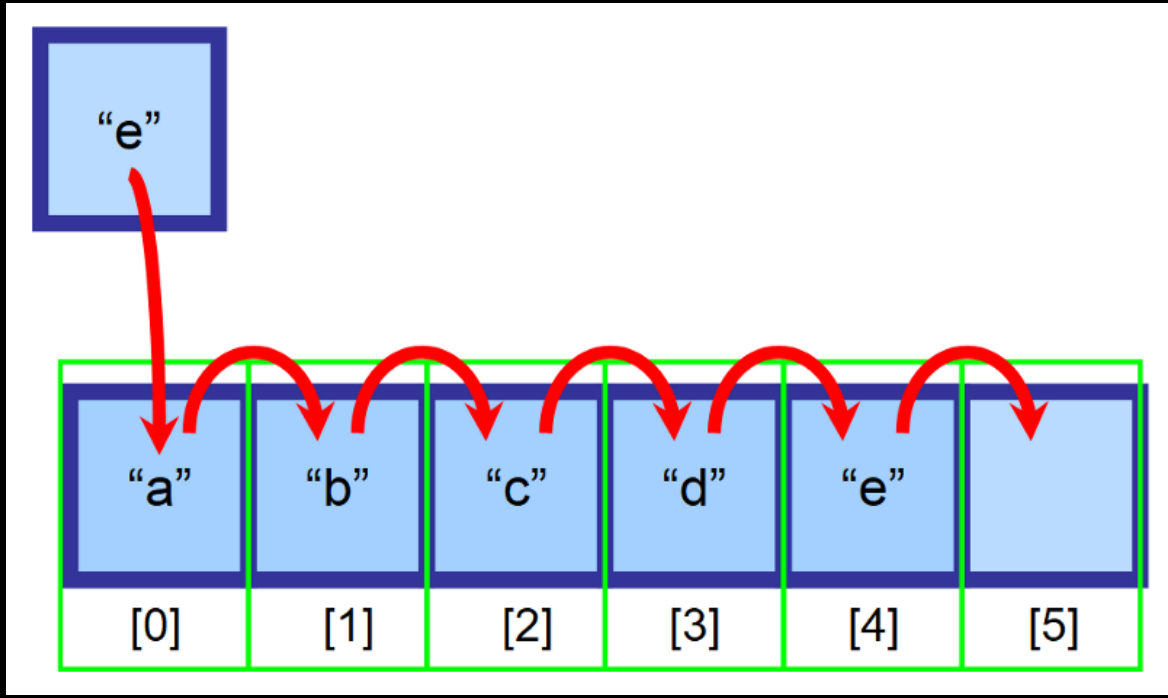
What is the fifth element in the array? 67

What is the result of

mean = (scores[0]+scores[1])/2;

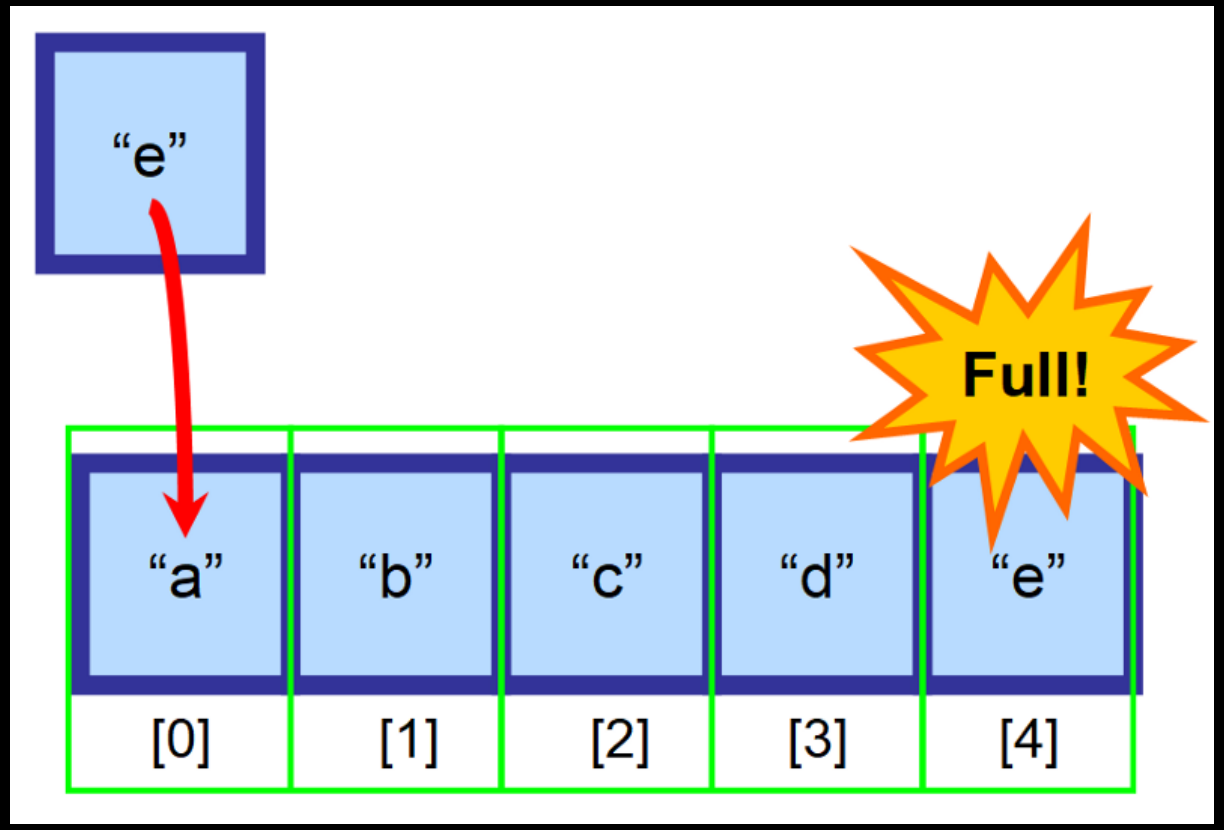
(79+87)/2 = 83

Can you workout the average of Quiz1and student0 with the following data??? 92,90



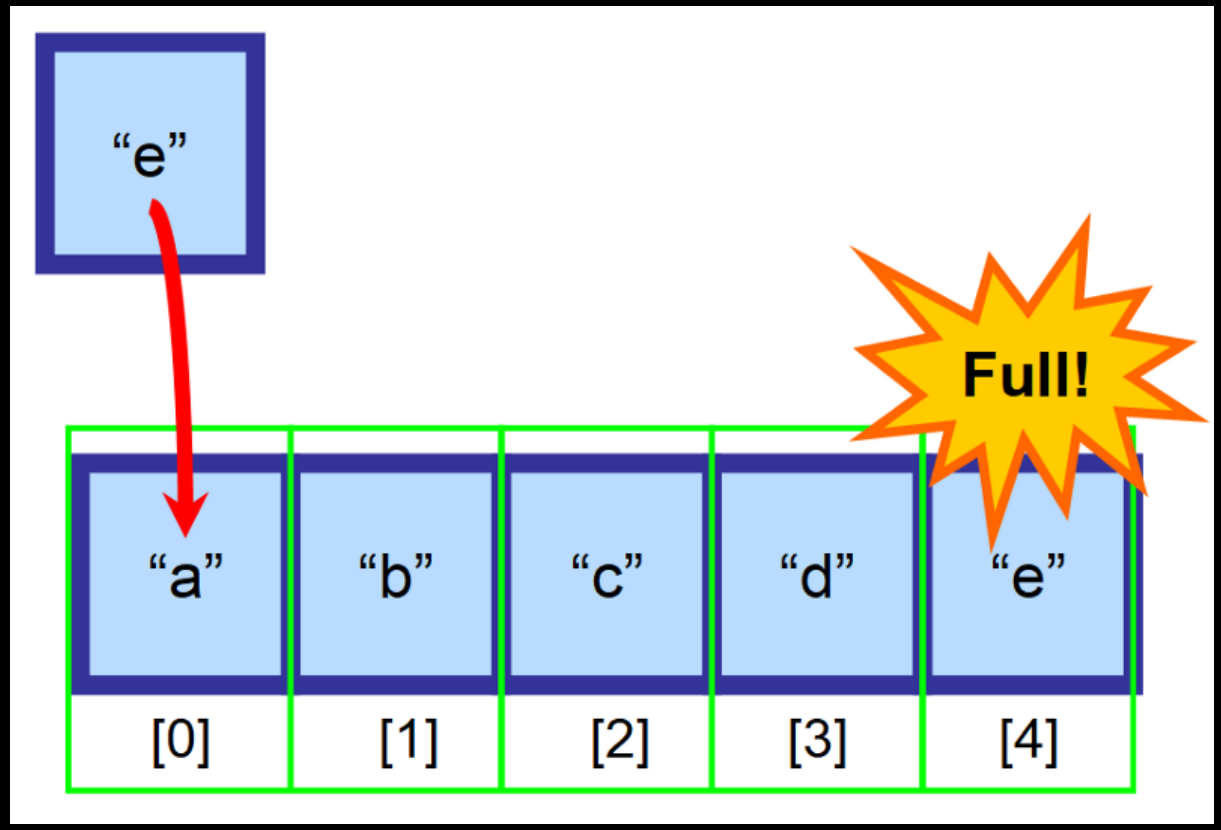
To insert an element into an array, all elements to the right side of the insertion site should be moved one index to the right. Only exception is the insertion at the end (if space is available)

https://www.freecodecamp.org/news/data-structures-101-arrays-a-visual-introduction-for-beginners-7f013bcc355a/



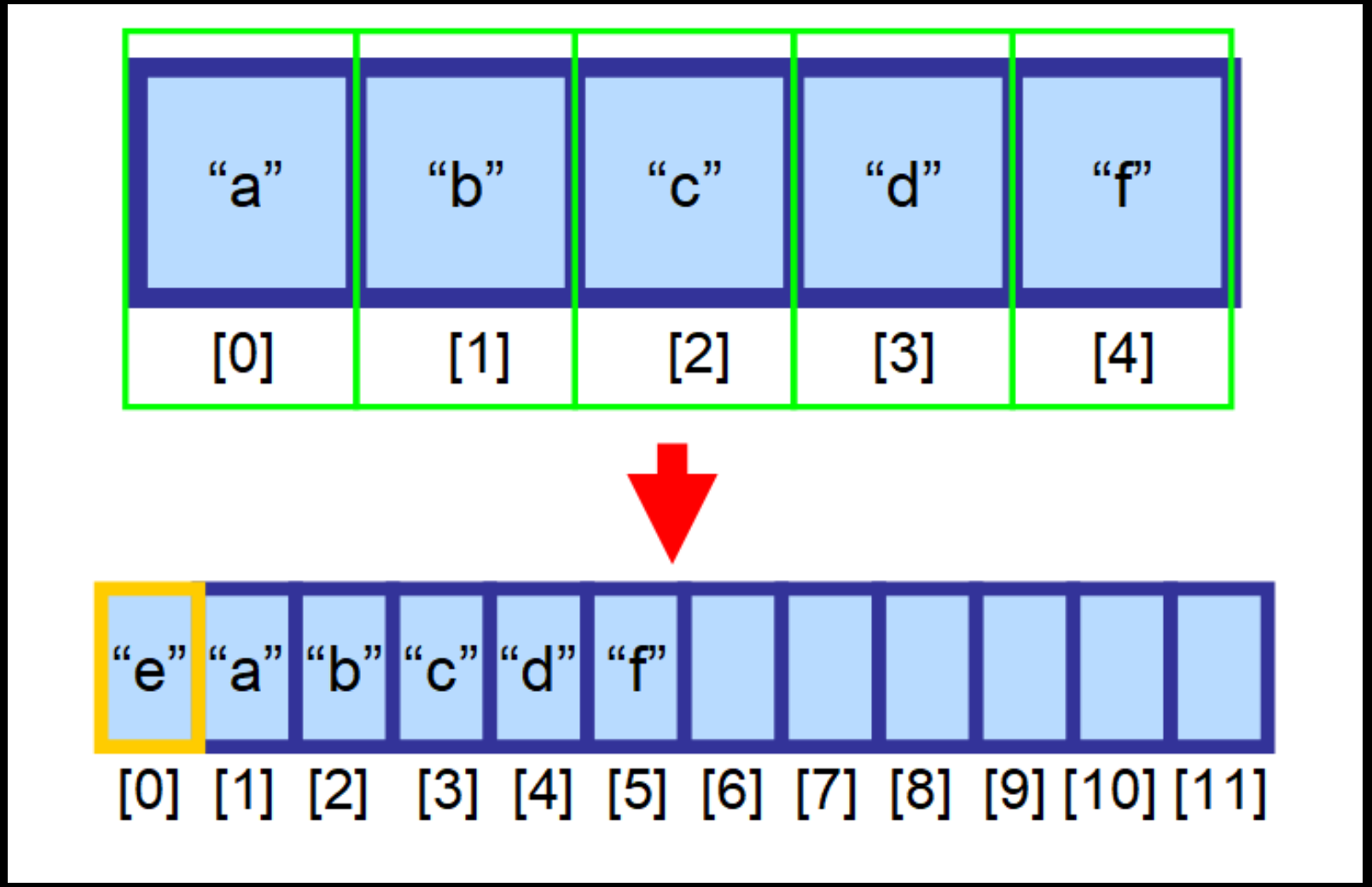
What if you must insert an element into an array that is full?

https://www.freecodecamp.org/news/data-structures-101-arrays-a-visual-introduction-for-beginners-7f013bcc355a/



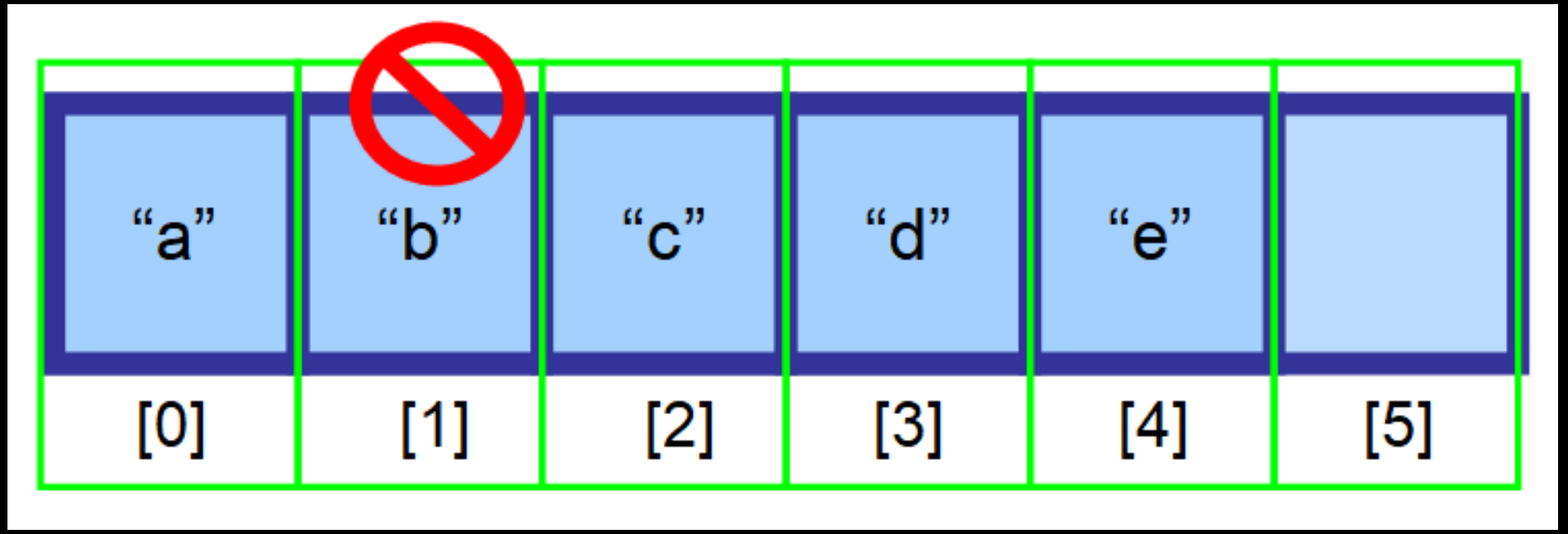
What if you must insert an element into an array that is full? You need to create a larger array and manually copy all the elements into the new array. A very expensive operation!!

https://www.freecodecamp.org/news/data-structures-101-arrays-a-visual-introduction-for-beginners-7f013bcc355a/



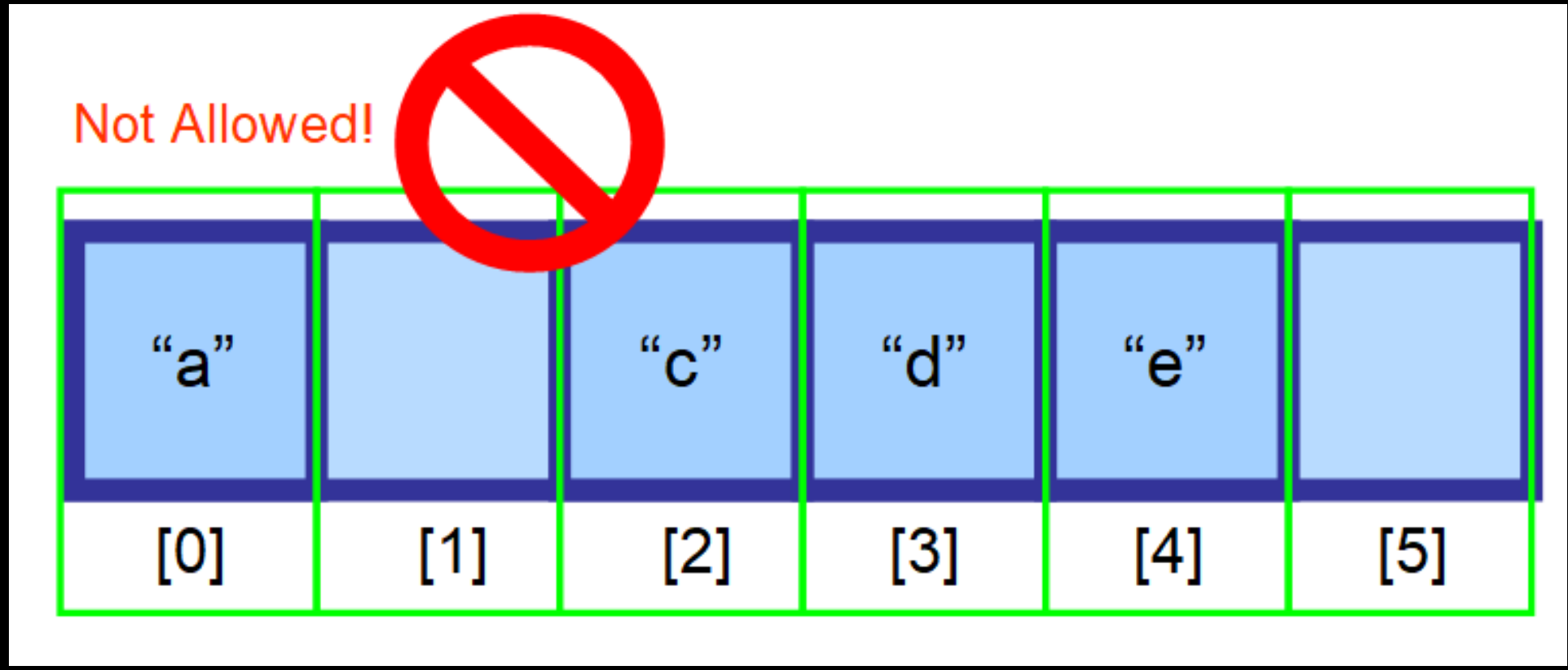
Like this. But this a very expensive operation in terms of time!!

https://www.freecodecamp.org/news/data-structures-101-arrays-a-visual-introduction-for-beginners-7f013bcc355a/



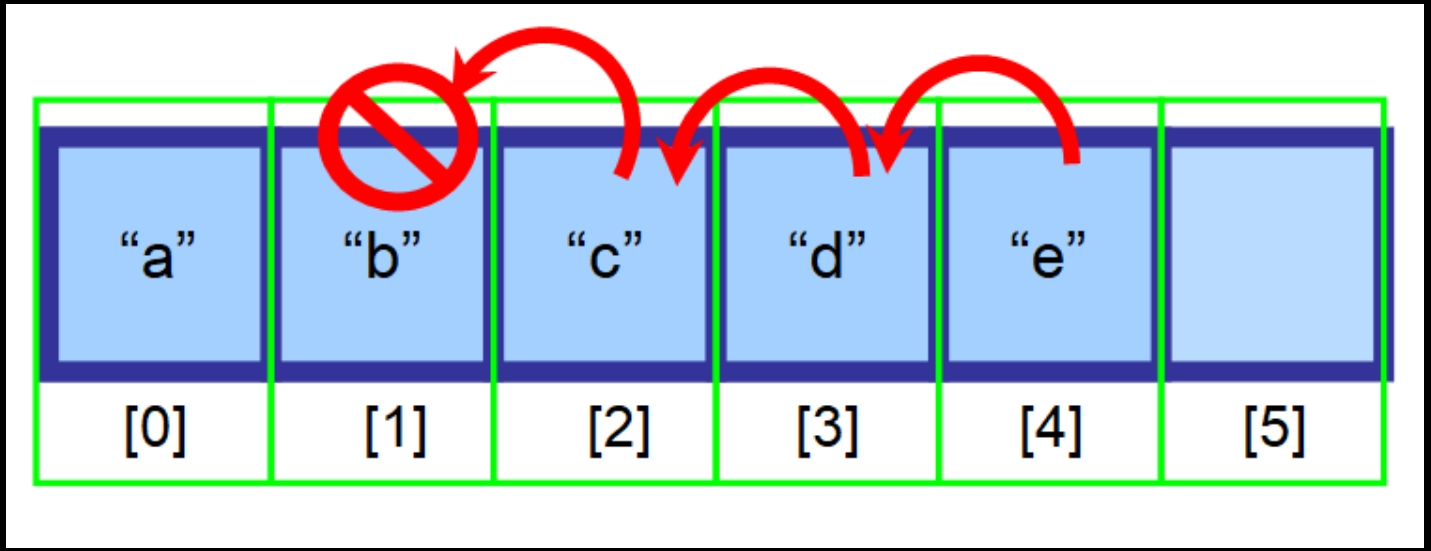
In case of deletion, you can't just delete an element and leave the space empty. Remember elements must be stored in contiguous memory location to maintain the efficiency of (random) access

https://www.freecodecamp.org/news/data-structures-101-arrays-a-visual-introduction-for-beginners-7f013bcc355a/



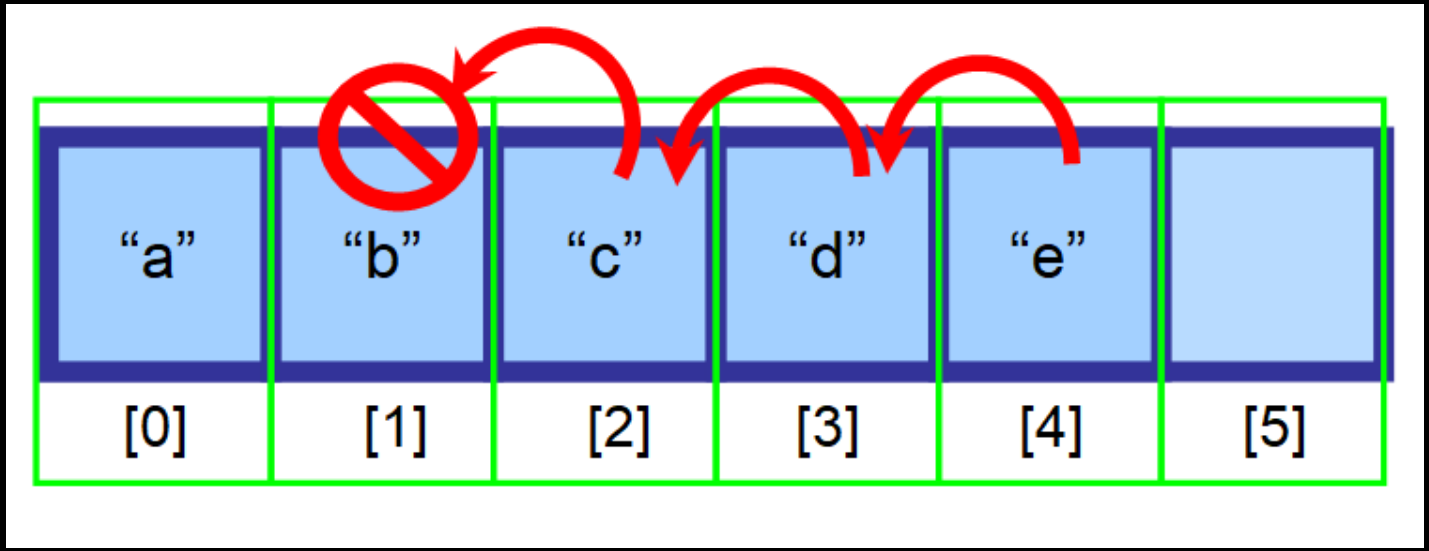
In that case, what do you think can be done?

https://www.freecodecamp.org/news/data-structures-101-arrays-a-visual-introduction-for-beginners-7f013bcc355a/

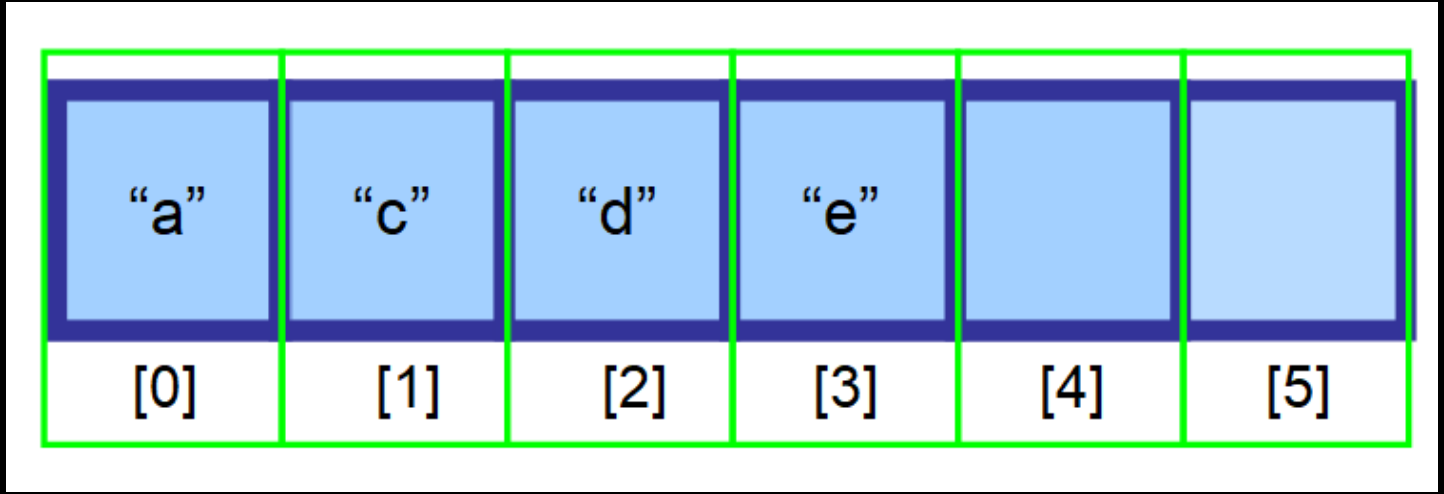


Move all the elements to the right of the element to be deleted by one index to the left

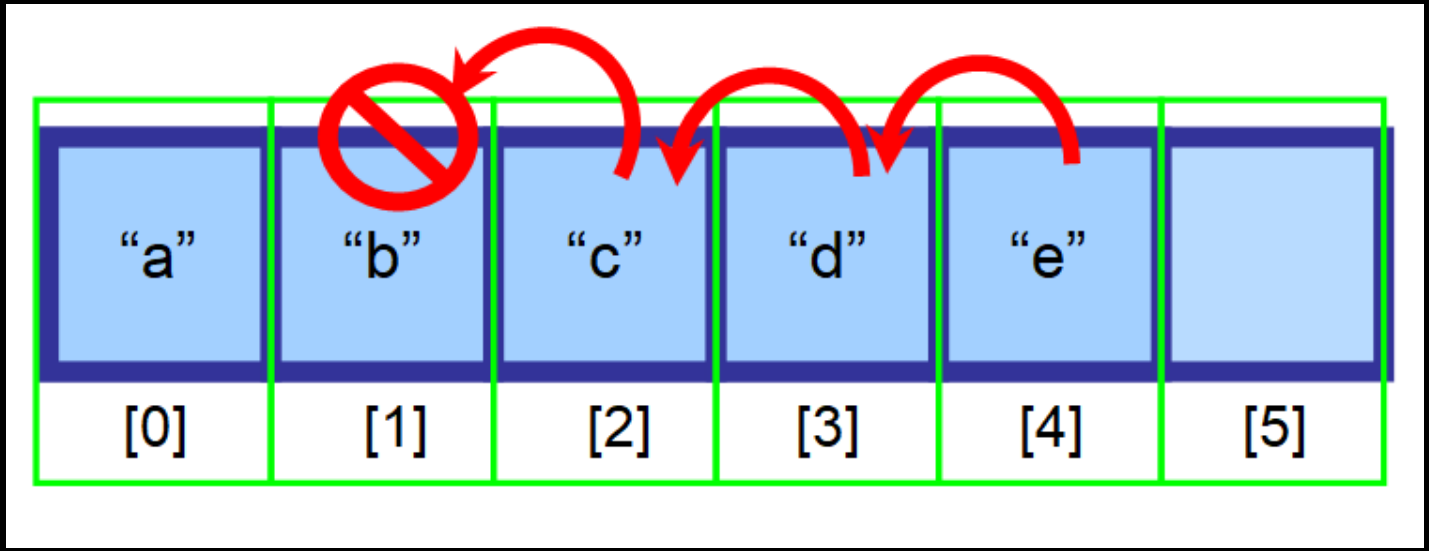
https://www.freecodecamp.org/news/data-structures-101-arrays-a-visual-introduction-for-beginners-7f013bcc355a/



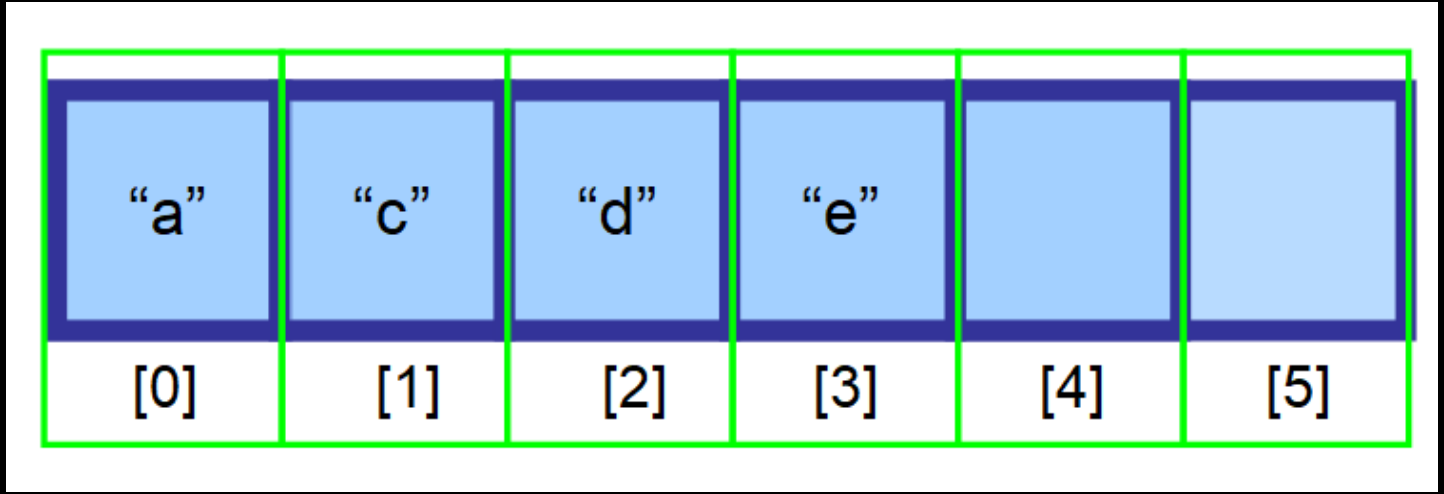
Move all the elements to the right of the element to be deleted by one index to the left



https://www.freecodecamp.org/news/data-structures-101-arrays-a-visual-introduction-for-beginners-7f013bcc355a/



When do you think deletion is very efficient?

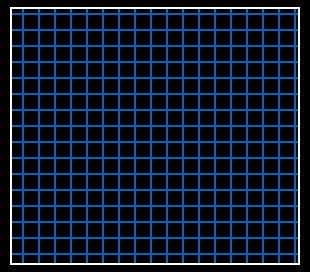


https://www.freecodecamp.org/news/data-structures-101-arrays-a-visual-introduction-for-beginners-7f013bcc355a/

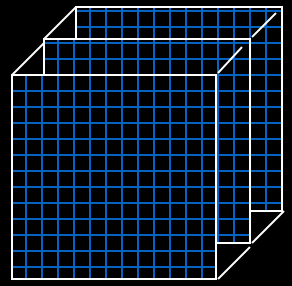
Multi Dimensional Arrays

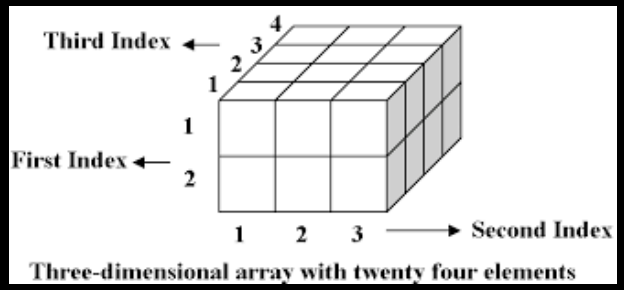
- An array can be declared in multiple dimensions

2 Dimensional (2D)

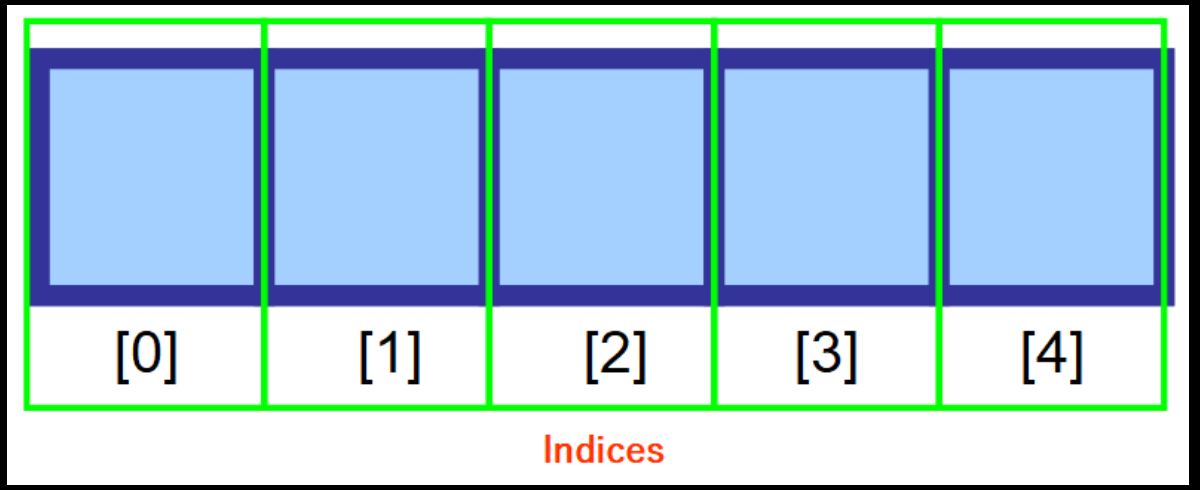


3 Dimensional (3D)



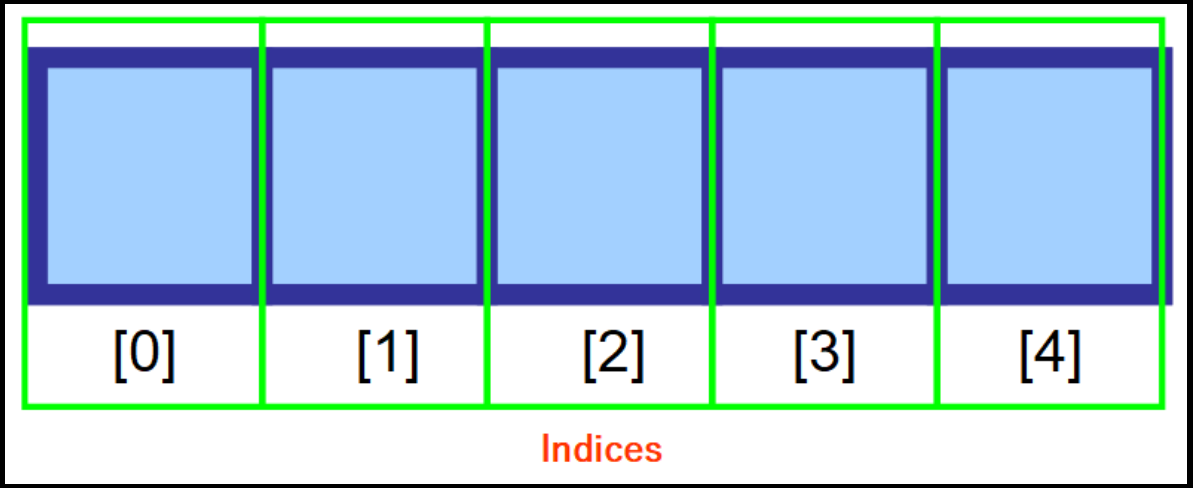


Three-dimensional array with twenty four elements



How to find an element in a 1D array?

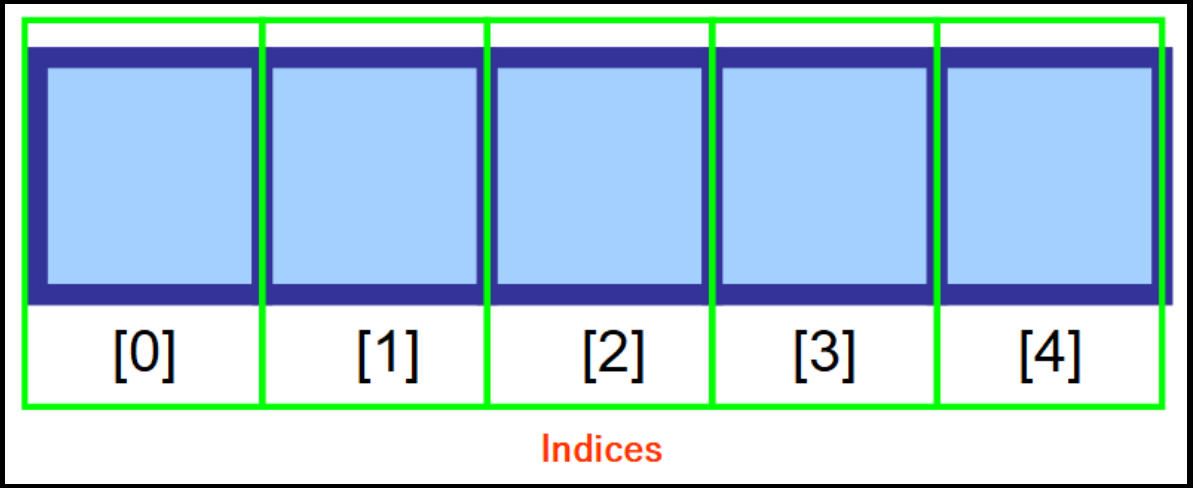
https://www.freecodecamp.org/news/data-structures-101-arrays-a-visual-introduction-for-beginners-7f013bcc355a/



How to find an element in a 1D array?

If you know where it's located use the indices (but it is very unlikely)

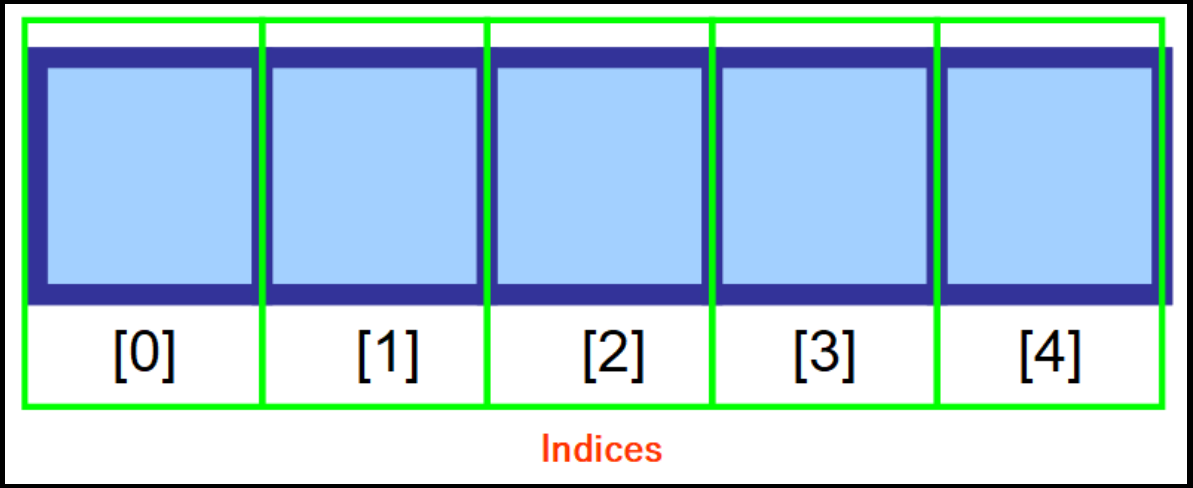
https://www.freecodecamp.org/news/data-structures-101-arrays-a-visual-introduction-for-beginners-7f013bcc355a/



How to find an element in a 1D array?

If you don't know where it's located, and the array is sorted there are algorithms to optimize the search (you have already seen binary search)

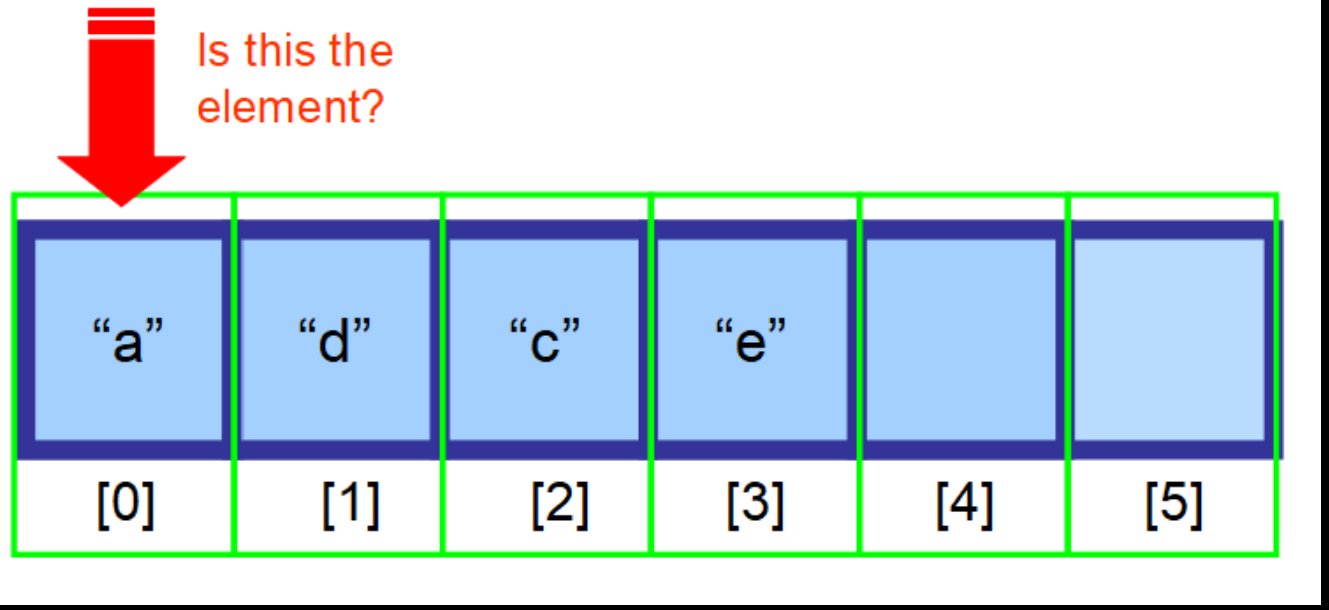
https://www.freecodecamp.org/news/data-structures-101-arrays-a-visual-introduction-for-beginners-7f013bcc355a/



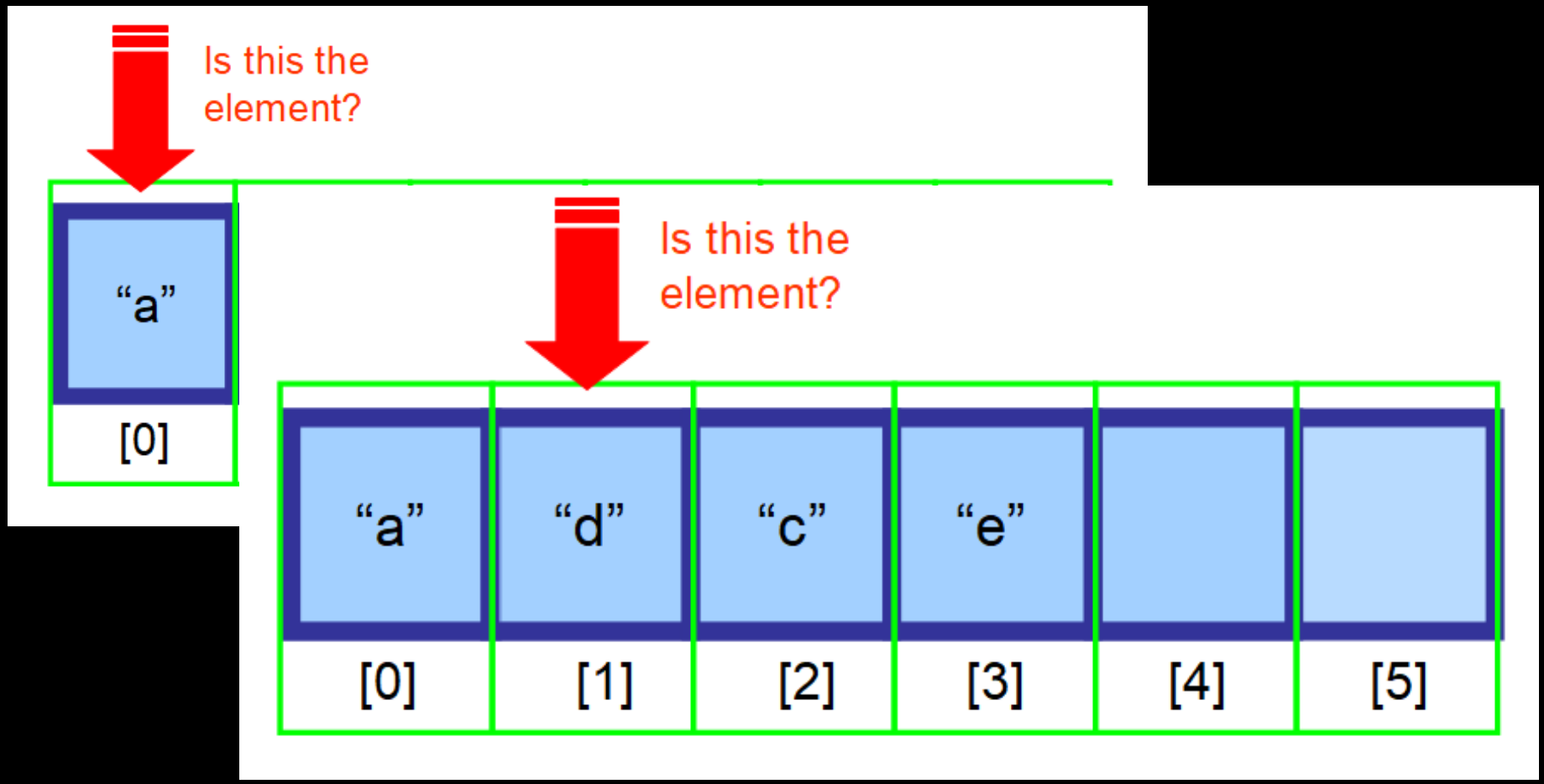
How to find an element in a 1D array?

If you don't know where it's located, and the array is not sorted then you need to search through every element in the array

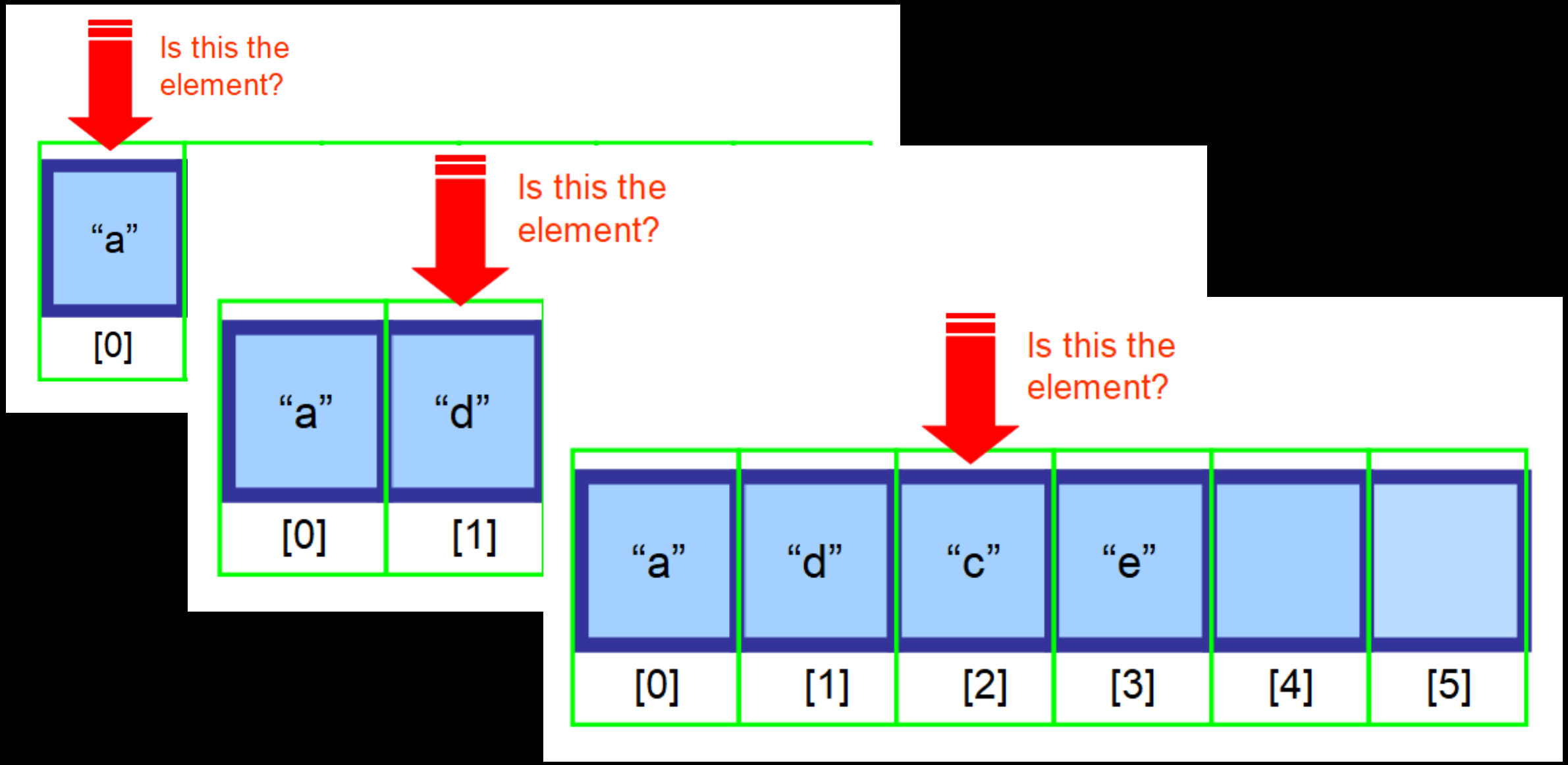
https://www.freecodecamp.org/news/data-structures-101-arrays-a-visual-introduction-for-beginners-7f013bcc355a/



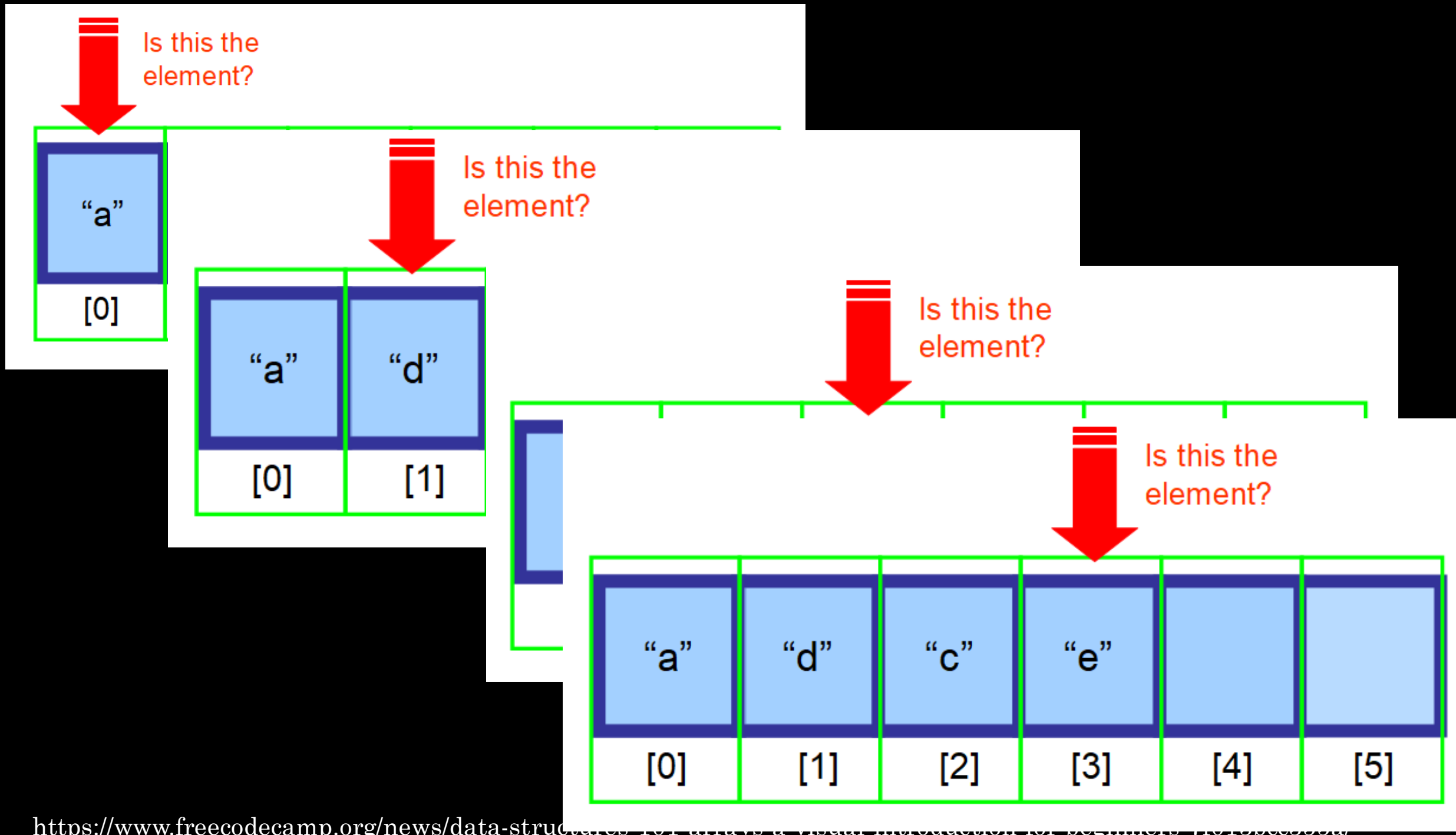
https://www.freecodecamp.org/news/data-structures-101-arrays-a-visual-introduction-for-beginners-7f013bcc355a/



https://www.freecodecamp.org/news/data-structures-101-arrays-a-visual-introduction-for-beginners-7f013bcc355a/



https://www.freecodecamp.org/news/data-structures-101-arrays-a-visual-introduction-for-beginners-7f013bcc355a/



https://www.freecodecamp.org/news/data-structures-101-arrays-a-visual-introduction-for-beginners-

NEXT TOPIC

SEARCHING