Arrays and Abstract Data Type in Data Structure (Wi...

Show Course Contents ⊕

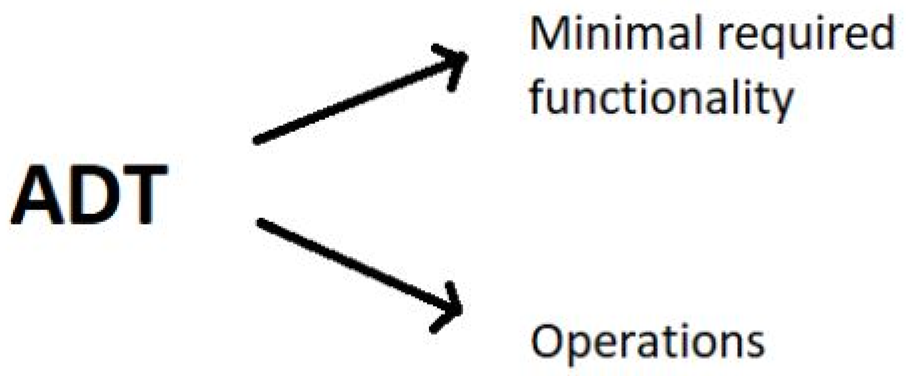
Overview Q&A Downloads Announcements

Arrays and Abstract Data Type in Data Structure (With Notes)

Today, we will learn about what an abstract data type is. This will just be an introduction. We'll implement these ideas in our next tutorial. Let's start with the basics.

Abstract Data Types and Arrays:

ADTs or abstract data types are the ways of classifying data structures by providing a minimal expected interface and some set of methods. It is very similar to when we make a blueprint before actually getting into doing some job, be it constructing a computer or a building. The blueprint comprises all the minimum required logistics and the roadmap to pursuing the job.



Array - ADT

An array ADT holds the collection of given elements (can be int, float, custom) accessible by their index.

1. Minimal required functionality:

We have two basic functionalities of an array, a get function to retrieve the element at index i and a set function to assign an element to some index in the array.

- get (i) – get element i
- set (i, num) – set element i to num.

2. Operations:-

We can have a whole lot of different operations on the array we created, but we'll limit ourselves to some basic ones.

- Max()
- Min()
- Search (num)
- Insert (i, num)
- Append (x)

Static and Dynamic Arrays:

- Static arrays – Size cannot be changed
- Dynamic arrays – Size can be changed

Quick Quiz- Code the operations mentioned above in C language by creating array ADT.

Hint: Use structures.

Memory Representations of Array:

- Elements in an array are stored in contiguous memory locations.
- Elements in an array can be accessed using the base address in constant time → O (1).
- Although changing the size of an array is not possible, one can always reallocate it to some bigger memory location. Therefore resizing in an array is a costly operation.

So, this was enough discussing the basics of an array ADT. It is now time to implement these data types through our codes. Even if you are not very familiar with these languages or need a quick brush-up, you can always move on to my easily accessible C or C++ playlist.

Thank you for being with me throughout. I hope you enjoyed the tutorial. If you appreciate my work, please let your friends know about this course too. Make sure to download the notes linked below. If you haven't checked out the whole playlist yet, move on to codewithharry.com or my YouTube channel to access it. See you all in the next tutorial, where we'll learn to construct an array as an abstract data type. Till then, keep learning.

Download Notes Here

Previous Next