

Array as An Abstract Data Type in Data Structures(With Notes)

 codewithharry.com/videos/data-structures-and-algorithms-in-hindi-7

In the last video, we learned what abstract data types are. In this video, we will be interested in implementing an array as an abstract data type. Giving it a quick revision, an abstract data type is just another data type as an int or float, with some user-defined methods and operations. It's a kind of customized data type.

Suppose we want to build an array as an abstract data type with our customized set of values and customized set of operations in a heap. Let's name this customized array myArray.

Let our set of values which will represent our customized array include these parameters:

- total_size
- used_size
- base_address

And the operations include operators namely,

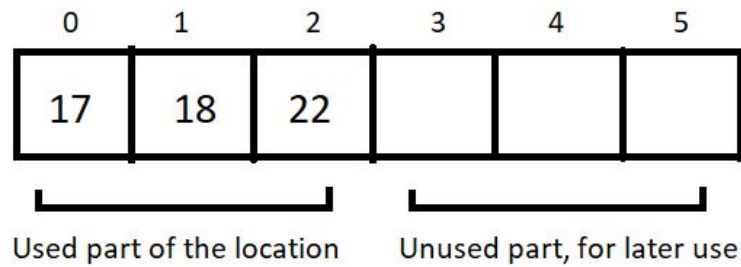
- max()
- get(i)
- set(i,num)
- add(another_array)

So, now when we are done creating a blueprint of the customized array. We can very easily code their implementation, but before that, let's first learn what these values and operations, we have defined, do:

Understanding the ADT above:

1. total_size: This stores the total reserved size of the array in the memory location.
2. used_size: This stores the size of the memory location used.
3. base_address: This is a pointer that stores the address of the first element of the array.

Let the below-illustrated array be an example of what we are talking about.



Here, the `total_size` returns 6, and the `used_size` returns 3.

We will keep the code implementation of the above ADT for the next tutorial. You can even give it a try on your editors. Use structs and define that set of values with proper data types. The next session will teach us that anyway.

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 about implementing the above array. Till then, keep learning.

[Download the OneNote Notebook I wrote here](#)