CCUBE
CODECHEF
Codechef Campus Chapter - ITMBU

DATA STRUCTURES

Preface

Consider this document as your cheat-sheet and the upcoming session to be a trial version (what I mean is, PLEASE COOPERATE). We are going to start our Data Structure sessions with the most basic one i.e, an Array.

Although we will have a quick recap of the concepts discussed below, it is recommended to read the document once before the session as I really want to keep the live session *'all about implementing'.*

Programming Language

We are going to use CPP as it is considered to be one of the fastest programming languages when it comes to Competitive Programming and also because **CPP is tough.** There are too many concepts you need to know about the **I am sure** using CPP during these sessions will help us all clear and build our basics more!

Content Pattern

It is possible that we might lose sight (and hope!) of important topics as we start to struggle with the basics of DS. Therefore, here is a list of topics (along with the reasons why you must learn it) that we will discuss foremost for every single Data Structure we pick and then dwell further.

1. Definition
   May sound boring but you should be able to give a one line explanation when a recruiter (or possibly a junior) asks you – what is an array?

2. What type of data does it handle and how?

3. Different ways of declaring and initializing that DS.

4. What operations does it perform?
   Example: With an array, the most basic operations we perform are insertion, deletion, traversal (iterating through the data structure), searching, sorting.

5. Its applications and (competitive) programming questions based on it.

Please note that we are not going to print patterns here. We will discuss some what about the syntax of CPP which was introduced in the latest version because it is new (and I like it!). Therefore, please brush up your programming teeth before you start with DS!

Also, TIME and SPACE Complexity is not an individual topic. It is attached to everything that comes under the paradigm that is computer engineering. So, each operation we discuss, we talk about its complexities **without a doubt!**

Session 1: Arrays.

…you have heard about them, you know them, you have probably used them many times, like MANY times, nevertheless…

1. Definition

- It is a physical data structure.
- An array is a **collection** of **similar type** of items stored in contiguous memory locations.
- Multiple items of similar type can be referenced via. only one reference variable using an array.
- i.e, you need only a single variable, say ‘table’ to reference an array containing multiplication table of 2.
  table -> {2, 4, 6, 8, 10}

Advantage

- Accessing any element is efficient in an array since data is stored in contiguous memory locations. Therefore, you can access any element via its index.

  i.e, if you need to know the product of 2*4, you just need to access the 4th element of the array ‘table’.

  table[3] = 8 …index starts from 0.

Drawback

- We will discuss drawbacks while discussing the next DS. So as to understand why we need an alternate data structure.

2. Data handling

- An array can only store homogenous data depending upon the data type of your array’s reference variable.
  i.e, if your reference variable ‘table’ is of type ‘int’, your array will only store integer type data.

- Implicit type conversion (conversion of one data type to another data type by the complier) is possible.

  For example:
  int characters [ ] = { ‘A’, ‘B’, ‘C’ } // gets converted to {65, 66, 67}
  Since the type of the array reference variable ‘characters’ is ‘int’.

3. Declaration and Initialization

- TO BE DISCUSSED IN LIVE SESSION

4. Operations

- Inserting an element
    - In the front
    - In the end
    - In the middle

- Deleting an element
    - In the front
    - In the end
    - In the middle

- Accessing an Element
    - Via. Index number

- Traversal
    - Iterating through the data structure

- Searching and Sorting
    - using pre-defined libraries.
    - By building functions.

5. Applications

- TO BE DISCUSSED IN LIVE SESSION
- For now, just remember that an array is used when you have similar type of data for example if all the data elements are integers or if you want to store a list of names (strings).

6. Practice Questions

- TO BE DISCUSSED IN LIVE SESSION

END NOTE

If you have any doubts regarding the content in this doc or any other related (or unrelated) topic please contact me I am as excited for this as you are. Any and every feedback is appreciated.

This was just a basic overview and there are high chances you are already familiar with a lot more than this. If that's the case (firstly, a round of applause) do attend the session as we will take a look at common practices of dealing with the data structure and solve problem questions based on them.

After the session, I will send you another document containing a link to all the questions/topic we cover so that you can have a look on your own because Programming is actually a self-taught thing. There exists no one who can teach you programming, just the ones who do it with you and you all learn in the process.

Good day!

Rinku Monani
monanira@gmail.com