

1.1 Arrays in Data Structure | Declaration, Initialization, Memory representation

Jenny's Lectures CS IT

1.1 In this video, we'll discuss how data is represented in memory, and the need for arrays. Faerie explains the need for arrays as we often have to process large amounts of data, and arrays allow us to store multiple values under one variable name. We'll cover how arrays can be declared and how data can be stored in memory. The declaration of arrays is language-specific, and in this video, we'll consider the syntax in the C language. The elements of the array are stored in consecutive or continuous locations, with their index starting from zero. The size of the array cannot be changed at runtime. We'll discuss how data is stored in memory, with the binary form of the data being converted and then stored. The formula to calculate the address of an element is the base address plus the index value multiplied by the size of the data type. We'll cover how to initialize the array at runtime using loops or predefined functions, and how to take data from the user and store it in an array. In the next video, we'll discuss how to insert data, traverse arrays, and perform different operations on 1D arrays. We'll also cover 2D arrays and how to access their values.

1.2. Array Operations - Traversal, Insertion | Explanation with C Program | DSA Course

You are also going to have to use the interposition function to insert the data at the beginning of the array. Finally, you are going to have to use the printf function to print the data at the end of the array.

1.3 Array Operations | Deletion from Array | Explanation with Code | Data Structure

And finally, we'll discuss the time complexity. value so delete the value at position four so now the updated array is now 5 and if you want to print this then you can write this value down at the bottom of the for loop and then you can print this value at the top of the for loop