

**University of Anbar
College of Computer Science and
Information Technology
Department of Computer Networks**

Arrays

Lecture 1

Dr. Ali Al-Kubaisi

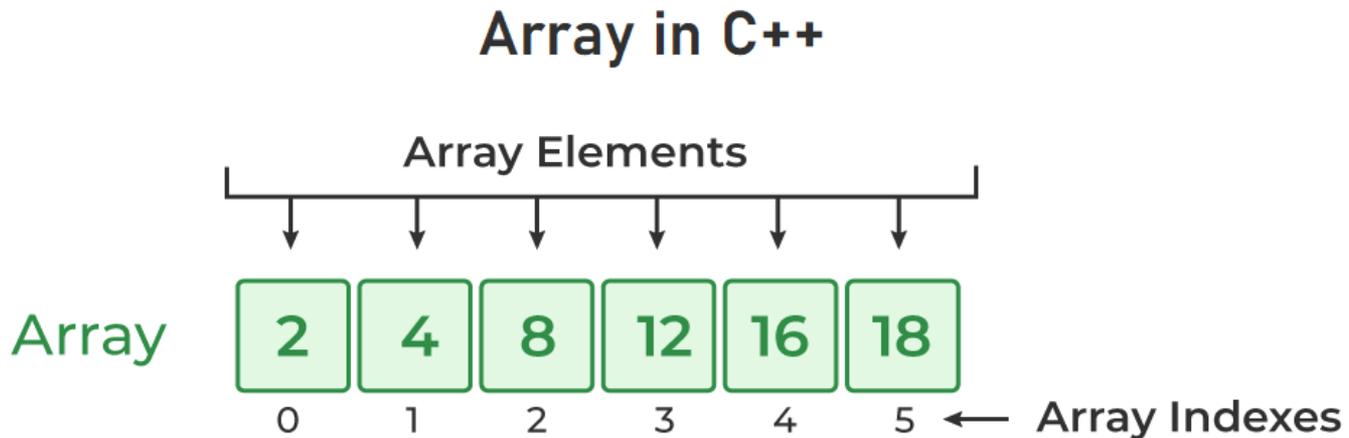
Asst.Lect. Abdulrahman Abbas Mukhlif

Outlines

- **C++ Arrays**
- **Create Array**
- **Initialize Array**
- **Access Array Elements**
- **Update Array Elements**
- **Loop Through an Array**

C++ Arrays

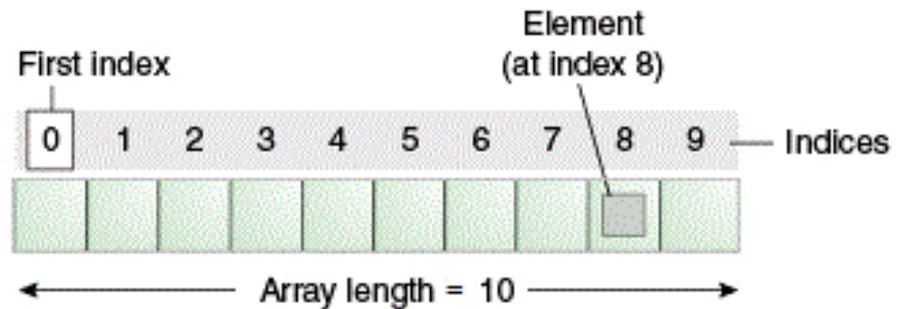
- Arrays are used to store multiple values in a single variable, instead of declaring separate variables for each value.



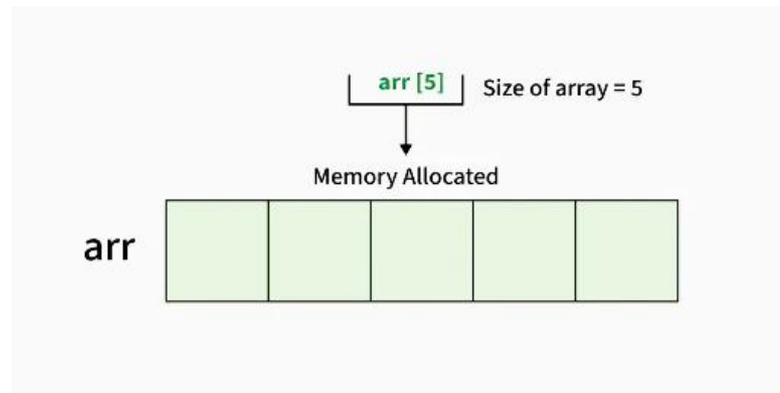
Create Array

- In C++, we can create/declare an array by simply specifying the data type first and then the name of the array with its size inside [] square brackets.

```
data_type array_name [size]
```

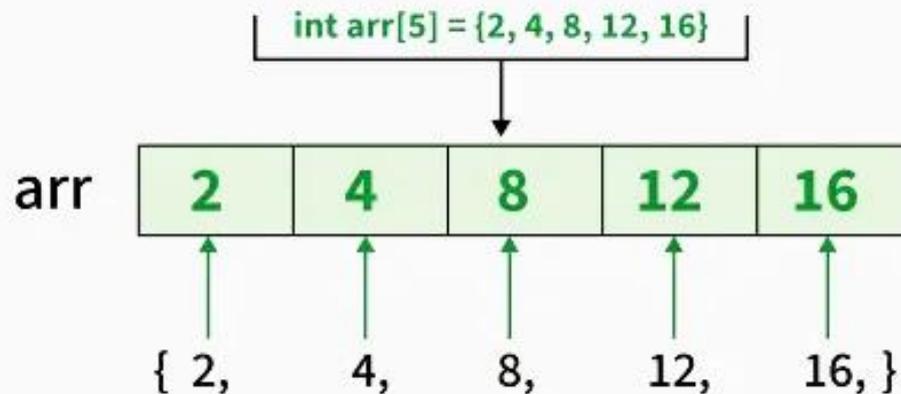


- Example `int arr[5];`
- This will create an array with name **arr** that can store **5 integers**.



Initialize Array

- Initialization means assigning initial values to array elements. We can initialize the array with values enclosed in curly braces '{ }' are assigned to the array.
- For example: `int arr[5] = {2, 4, 8, 12, 16};`



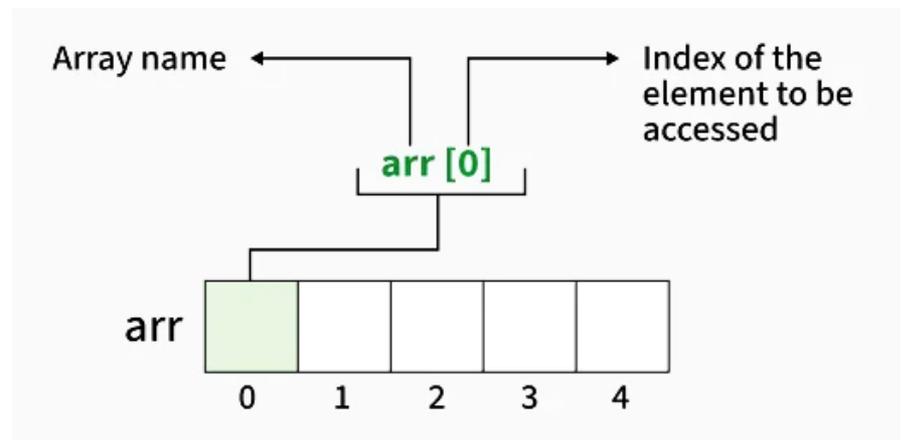
Access Array Elements

- Elements of an array can be accessed by their position (called index) in the sequence. In C++, indexes of an array starts from 0 instead of 1. We just have to pass this index inside the [] square brackets with the array name as shown:

```
array_name[index];
```

- It is important to note that index **cannot** be **negative** or **greater than size of the array minus 1**. ($0 \leq \text{index} \leq \text{size} - 1$).

```
int arr[] = {2, 4, 8, 12, 16};  
  
// Accessing fourth element  
cout << arr[3] << endl;  
  
// Accessing first element  
cout << arr[0];
```

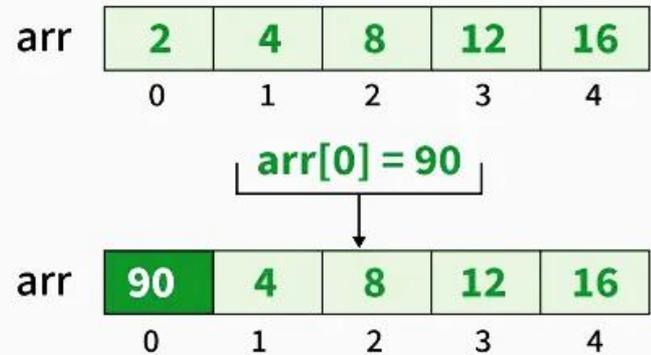


Update Array Elements

- To change the element at a particular index in an array, just use the = assignment operator with new value as right hand expression while accessing the array element.

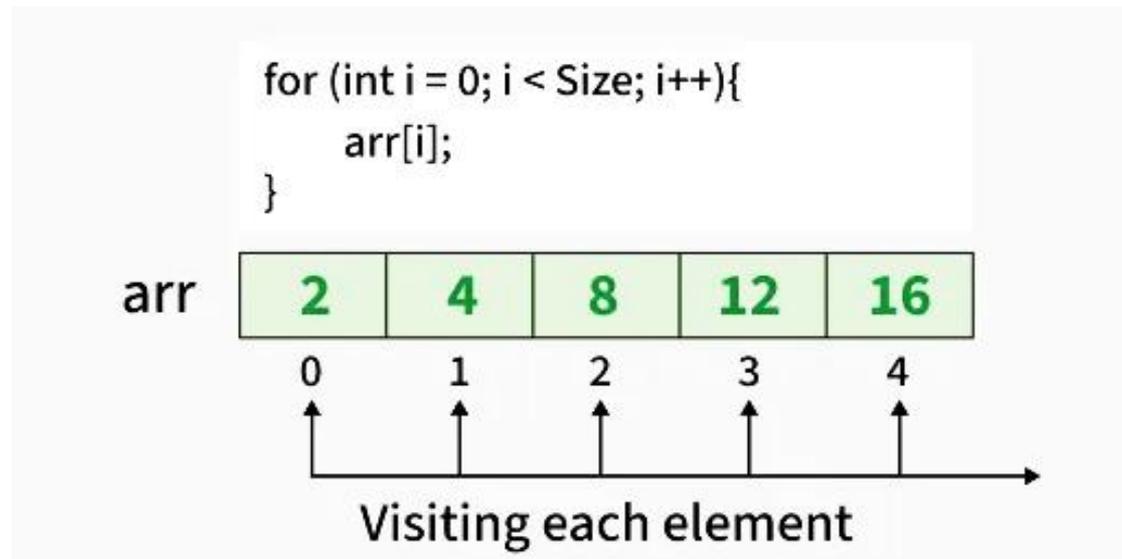
```
array_name[index] = value;
```

```
int arr[] = {2, 4, 8, 12, 16};  
    // Updating first element  
arr[0] = 90;  
cout << arr[0] << endl;  
    // Updating second element  
arr[1] = 90;  
cout << arr[0];
```

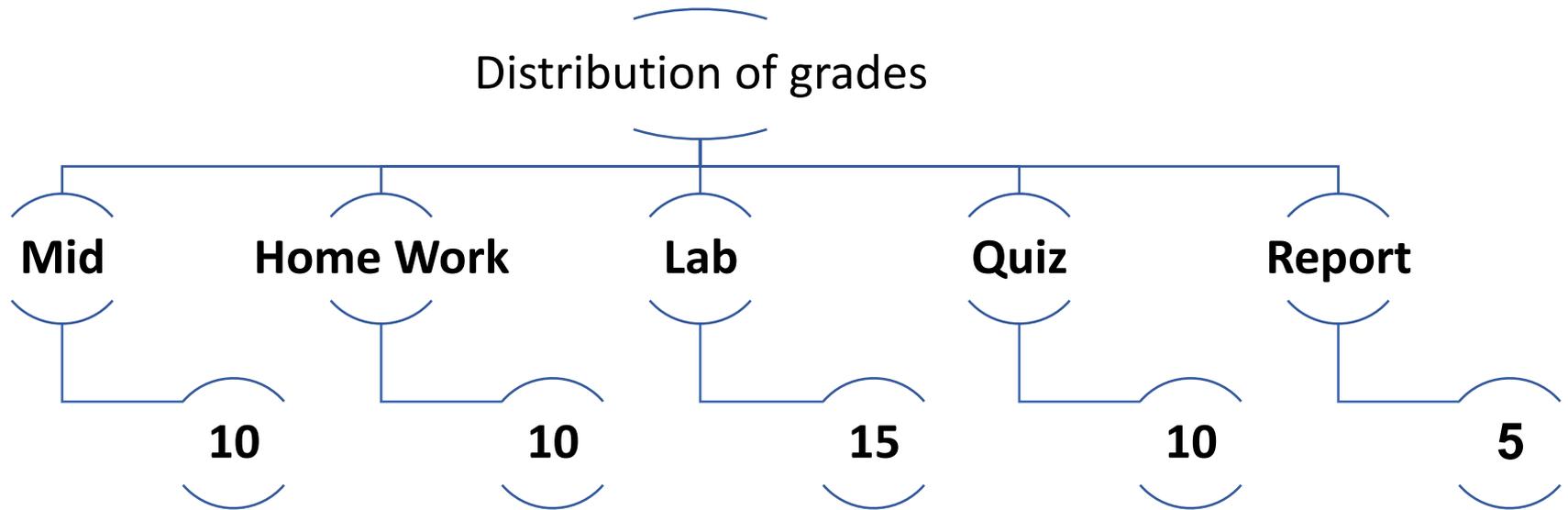


Loop Through an Array

- For iterating through an array, we create a loop that iterates the same number of times as there are elements in the array. For that, we will create a loop variable that starts from **0** (as arrays in C++ are 0-indexed), increments by one in each iteration and goes till it is **less than the size of the array**.
- We can use any loop of our choice. Here, we are using **for** loop



Distribution of grades



Learning and Teaching Resources

- The C++ Programming Language (4th Edition) by by Bjarne Stroustrup
- <https://www.learncpp.com/>
- <https://www.w3schools.com/CPP/default.asp>