Arrays

Terms

Arrays
Array elements
Bubble sort algorithm
Linear search algorithm
Pointers
Unpacking arrays

Summary

- An *array* is a collection of items stored in sequence.
- We can access the items (also called *elements*) in an array using an index. The index of the first element in an array is 0.
- An array can be unpacked into separate variables.
- Using the **size()** function in the C++ Standard Library (STL) we can determine the size of an array.
- When passing an array to a function, we should always pass its size as well. The reason for this is that array parameters are interpreted as *pointers* (memory addresses). We cannot use a memory address to determine how many elements exist in an array.
- In C++, we cannot assign an array to another. To copy an array, we need to copy individual elements using a loop.
- Similarly, arrays cannot be compared for equality. To compare two arrays, we have to compare their individual elements.
- There are many algorithms for searching and sorting arrays. These algorithms differ based on their performance and memory usage.


```cpp
// Creating and initializing an array
int numbers[] = { 1, 2, 3 };
string names[5];

// Accessing elements in an array
numbers[0] = 10;
cout << numbers[0];

// Determining the size of an array
auto size : = std::size(numbers);

// Unpacking arrays
auto [x : int , y : int , z : int ] = numbers;

// Multi-dimensional arrays
int matrix[2][3] = {
        { 11, 12, 13 },
        { 21, 22, 23 }
};

matrix[0][0] = 10;
```

