

Data Structures

By: Aaryan Vinod Kumar

A data structure is a systematic way of organizing, managing, and storing data so that it can be accessed and modified efficiently. Data structures are the foundation of writing efficient algorithms and solving complex computational problems.

They define:

- How data is laid out in memory
- How operations like insertion, deletion, traversal, and searching are performed

Data structures play a pivotal role in problem-solving and are fundamental to the design of efficient algorithms. A strong understanding of data structures is essential, as they form the backbone of numerous real-world applications. Gaining proficiency in this area is advantageous, irrespective of whether one aspires to pursue a career in the technology sector or not.

Types of Data Structures

Data structures can broadly be divided into two categories:

1. Primitive Data Structures

Primitive data types are the basic data types provided by programming languages as building blocks. They directly operate upon the memory locations. Think of these like your cooking utensils (you don't need to make your own knife before you start cooking, for example.)

Broadly, all programming languages have the following Data Structures:

Name	Description	Example
<code>int</code>	Integer Numbers	1, 2, 3, -9
<code>char</code>	Any Character	'A', 'b', '100'
<code>float</code>	Decimal Numbers	1.99..., 5.14
<code>bool</code>	True or False	true, false

2. Non-Primitive Data Structures

These are derived data types that use primitive types and organize them in specific ways to perform complex operations. These are like cooking something new from what you already have, if we follow the same example, so making a dish from the veggies using the utensils.

Non-Primitive types are classified as:

a. Linear Data Structures

In linear data structures, data elements are arranged sequentially (one-by-one), and each element is connected to the next.

Examples:

- **Array:** Collection of similar elements stored in contiguous memory locations.
Example: `int arr[5] = {1, 2, 3, 4, 5};`
- **Linked List:** Elements (nodes) are linked using pointers. Each node contains data and a reference to the next node.
- **Stack:** Follows the Last In First Out (LIFO) principle. Operations are usually `push` (insert) and `pop` (remove).
- **Queue:** Follows the First In First Out (FIFO) principle. Supports `enqueue` and `dequeue` operations.

b. Non-Linear Data Structures

In non-linear data structures, elements are not arranged sequentially. These structures are used to represent complex relationships.

Examples:

- **Tree:** Hierarchical structure. Each node may point to multiple child nodes.
Common types: Binary Tree, Binary Search Tree
- **Graph:** Set of nodes (vertices) connected by edges. Can be directed or undirected.

If you don't understand much of this right now, no need to worry, I had the same confused unga-bunga vibe when I first tried to make sense of all this. Just get a general sense of what Data Structures are, and pick a programming language to practice these in.

Wait, what's a programming language though?

Programming Language: A programming language is a formal set of instructions used to produce a wide range of outputs, such as applications, algorithms, and data manipulations. It enables humans to communicate with computers and provide them with a set of rules for performing tasks.

Think of a programming language as a medium to tell a computer to perform an action. Since computers only understand 0s and 1s, and writing in 0s and 1s is really hard, hence we use programming languages.

All of the programmes provided would be in C++ (cpp) language, installation steps for which are given at the end.

Note: Data Structures are **independent** of programming language. Most of the topics are **abstract** and hence, can be implemented in any language, but be different in each. Here, you just need to focus on the concept, not how to write it in a certain language.

How to install C++?

Windows:

1. Install Visual Studio Code

1. Download VS Code:

Visit the official [Visual Studio Code website](https://code.visualstudio.com/) and click on the "Download for Windows" button.

2. Run the Installer:

Once downloaded, open the installer file.

3. Setup Wizard:

Follow the prompts in the setup wizard:

- Accept the license agreement.
- Choose the installation location.
- Select additional tasks (like creating a desktop icon).

4. Complete Installation:

Click "Install" to begin the installation process. Once completed, click "Finish" to launch VS Code.

2. Install MinGW (Minimalist GNU for Windows)

1. Download MinGW:
Go to the MinGW website and download the latest installer.
2. Run the Installer:
Open the downloaded `mingw-get-setup.exe` file.
3. Install MinGW:
In the installation manager:
 - 3.1. Click "Install" to proceed.
 - 3.2. After installation, the MinGW Installation Manager will open.
4. Select Packages:
In the Installation Manager:
 - 4.1. Mark the following packages for installation:
 - 4.1.1. `mingw32-base`
 - 4.1.2. `mingw32-gcc-g++`
 - 4.2. Go to the "Installation" menu and click "Apply Changes".
5. Set Environment Variables:
To use MinGW from the command line:
 - 5.1. Right-click on "This PC" or "My Computer" and select "Properties".
 - 5.2. Click on "Advanced system settings".

- 5.3. In the "System Properties" window, click on "Environment Variables".
- 5.4. Under "System variables", find and select the "Path" variable, then click "Edit".
- 5.5. Click "New" and add the path to MinGW's `bin` directory (e.g., `C:\MinGW\bin`).
- 5.6. Click "OK" to close all dialogs.

Verify Installation:

Open Command Prompt and type:

```
g++ --version
```

If installed correctly, it will display the version of GCC.

Mac:

1. Install Visual Studio Code

1. Download VS Code:
Navigate to the [Visual Studio Code website](#) and download the macOS version.
2. Install the Application:
Open the downloaded `.zip` file, which will extract the `Visual Studio Code.app`.
3. Move to Applications:
Drag the `Visual Studio Code.app` into your "Applications" folder.
4. Launch VS Code:
Open "Applications" and double-click on "Visual Studio Code" to launch it.

2. Install MinGW (via Homebrew)

MinGW is primarily designed for Windows. On macOS, you can install `mingw-w64`, which provides similar functionality.

Install Homebrew (if not already installed):

Open Terminal and run:

```
/bin/bash -c "$(curl -fsSL  
https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
```

Install mingw-w64:

After Homebrew is installed, run:

```
brew install mingw-w64
```

Verify Installation:

Check the installation by typing:

```
x86_64-w64-mingw32-gcc --version
```

This should display the version of the installed compiler.