**DATA STRUCTURES**

A data structure is a way of organizing and storing data in a computer so that it can be accessed and used efficiently. There are many different types of data structures, each with its own strengths and weaknesses, but they all share the common goal of enabling efficient data access and manipulation.

Here are some examples of common data structures:

* **Arrays**: An array is a collection of elements of the same type stored in contiguous memory locations. The elements can be accessed using an index that starts at 0. Arrays are useful when you need to store a fixed number of elements and want to access them quickly. For example, an array can be used to store the grades of a class of students.
* **Linked lists**: A linked list is a collection of nodes that each contain a value and a reference to the next node in the list. Linked lists are useful when you need to store a variable number of elements and don't want to waste memory by pre-allocating a fixed size. They can be used to implement data structures like stacks and queues.
* **Trees**: A tree is a hierarchical data structure that consists of nodes connected by edges. Each node has a value and can have zero or more children nodes. Trees are useful when you need to store data in a hierarchy, such as the file system on a computer or the organizational structure of a company.
* **Hash tables**: A hash table is a data structure that stores key-value pairs in an array. The keys are hashed to produce an index into the array, which is used to store the corresponding value. Hash tables are useful when you need to look up values quickly based on a key, such as a dictionary.
* **Graphs**: A graph is a collection of nodes and edges that connect them. Graphs are useful when you need to represent relationships between objects, such as social networks or road networks.

These are just a few examples of the many different data structures that exist. Each one has its own strengths and weaknesses, and choosing the right one for a particular problem can have a big impact on the efficiency of your code.