Abstract Data Types in Data structures

An abstract data type is an abstraction of a data structure that provides only the interface to which the data structure must adhere.
The interface does not give any specific details about something should be implemented or in what programming language.

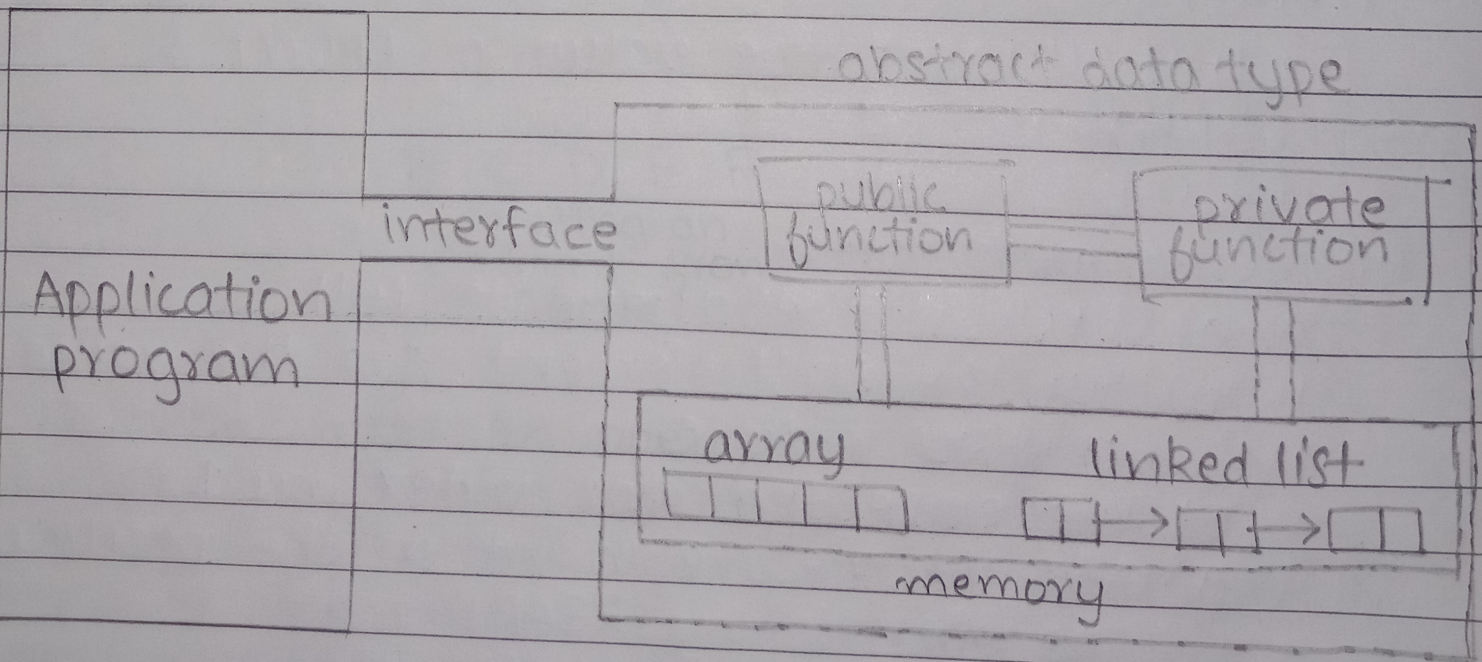
In other words, we can say that abstract data types are the entities that are definitions of data & operations but do not have implementation details.
In this case, we know the data that we are storing & the operation that can be performed on the data, but we don't know about the implementation details.
The reason for not having implementation details is that every programming language has a different implementation strategy. For example, a C data structure is implemented using Structures while a C++ data structure is implementation using object & classes.

For example, a list is an abstract data type that is implemented using a dynamic array & linked list. A queue is implemented using linked-list based queue, array based queue & stack based queue. A map is implemented using Tree map, hash map, or hash table.

* Abstract Data Type Model :-

i) Abstraction : It is a technique of hiding the internal details from the user & only showing the necessary details to the user.

ii) Encapsulation : It is a techniaue of combining the data & the member function in a single unit is known as encapsulation.



The figure contains ADT model. There are 2 types of models in ADT model, i.e., the public function & the private function.

The ADT models also contains the data structures that we are using in a program.

In the above model, first encapsulation is performed, i.e., all the data is wrapped in a single unit, i.e., ADT. Then the abstraction is performed means showing the operations that can be performed on the data structure & what are the data structures that we are using in a program.