

ABSTRACT DATA TYPES

**DATA STRUCTURE &
ALGORITHM
PCC-CS301**

**ARKAPRATIM GHOSH
13000121058
CSE-A**





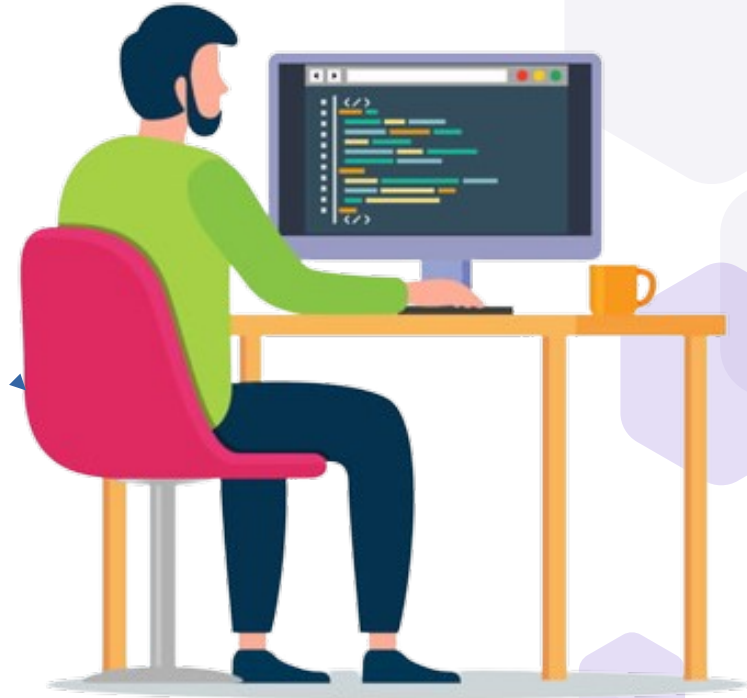
CONTENT

INTRODUCTION

- ⦿ What is a datatype?
- ⦿ User-defined datatype

ABSTRACT DATA TYPES

- ⦿ DEFINITION
- ⦿ IMPLEMENTATION
- ⦿ ADVANATGES





1

INTRODUCTION

Let's start with the basics



WHAT IS A DATATYPE ?

- ◆ A classification that indicates what a variable or object can hold
- ◆ Defines a certain **domain** of values
- ◆ Defines **operations** allowed on those values

EXAMPLE

int type - Takes integer values. Allows operations like addition, subtraction, multiplication, division etc.



USER DEFINED DATATYPE

The operations of user defined datatypes are specified by users

Examples:
structure, union,
enumerations

By using structure we are defining our own datatype by combining other datatypes

```
struct student{  
    int class;  
    int roll;  
}
```





2

ABSTRACT DATATYPES (ADTs)

What's so abstract about it ?



DEFINITION AND IMPLEMENTATION

ADTs are like user defined datatypes which defines operations on values using **functions** without specifying what is there inside the functions and how operations are performed

EXAMPLE: Stack ADT - consists of elements of same type arranged in a sequential order, implemented by using arrays and linkedlists

Operations: initialize()- initialising it to be empty.

push()- insert an element into the stack.

pop()- remove an element from the stack.

isEmpty()- checks if the stack is empty or not.

isFull()- checks if the stack is full or not.

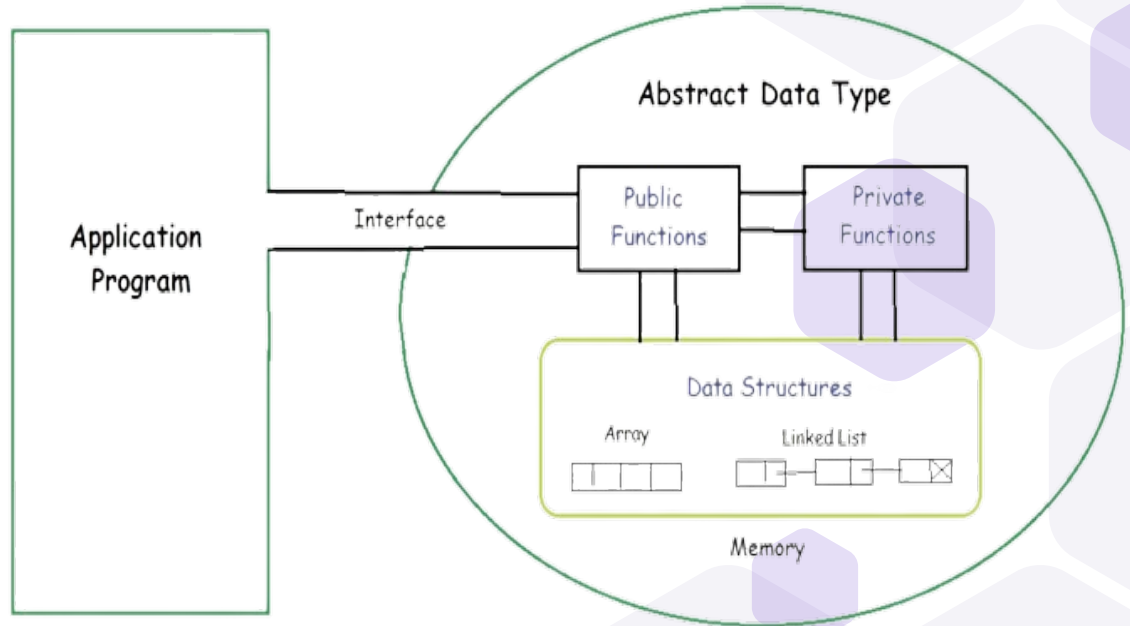


ADVANTAGES

Acts as a black box and interface.

A user's actions are not affected if the underlying implementation of an ADT is changed as the client program has nothing to do with it.

ADT provides
ABSTRACTION



THANK YOU

