

Habib University
shaping futures

CS 201 Data Structure II (L2 / L5)

Abstract Data Type

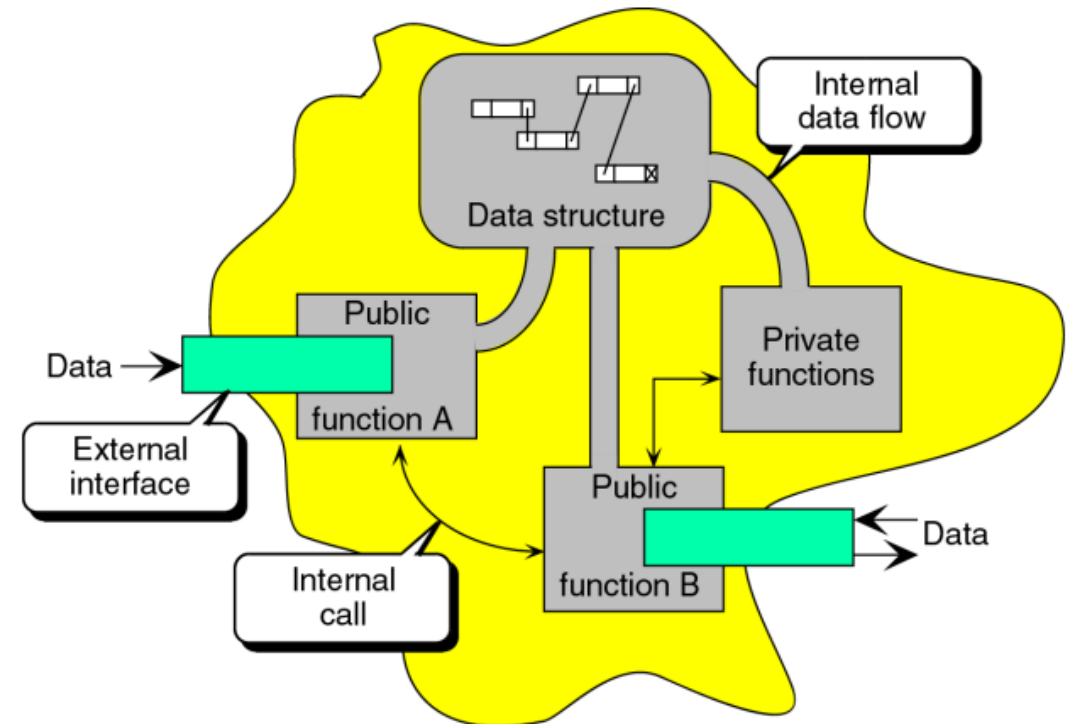
Muhammad Qasim Pasta

qasim.pasta@sse.habib.edu.pk

Why do we need Data Structures?

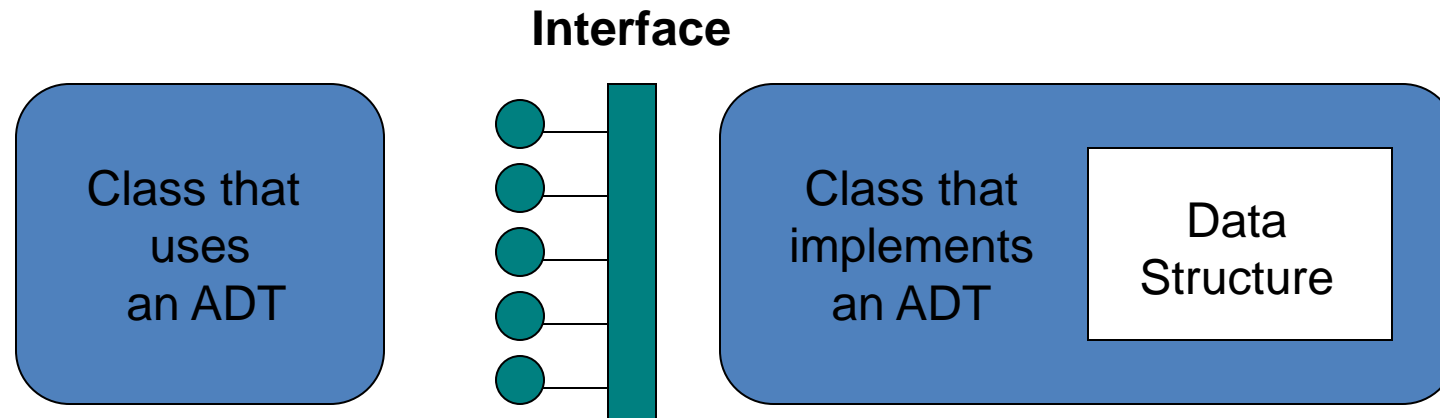
Abstract Data Type / Interfaces

- **Abstract Data Type (ADT)**
defines WHAT not HOW
 - defines the operations that can perform on the data
 - not define how to perform those operations
- a data type that has values and operations that are not defined in the language itself



Data Structures

- a programming construct used to implement an Abstract Data Type





Example ADTs

- Stack
 - operations?
- Queue
 - operations ?
- List
 - operations ?
 - `get(i)`, `set(x, i)`, `add(x)`, `remove(i)`, `size()`



Performance of Data Structures

Three things matter:

- **Correctness**
 - Should correctly implement the ADT/interface
- **Time Complexity**
 - Running time should be as small as possible
 - Three kinds of running time: worst-case, amortized, expected
- **Space Complexity**
 - Should use as little memory as possible

Reading

- Chapter #1

