

Data Structure and Algorithms

Affefah Qureshi

Department of Computer Science

Iqra University, Islamabad Campus.

Data Types

- Primitive

- Defines a certain domain of Values
- Define operation allowed on those values
- Example:
 - Data Values: can take values with decimal points
 - Operations: Addition, subtraction and multiplication but not remainder

- User Defined

- Domain and operations are not defined by language but by user
- Example:

```
struct person{  
    char name;  
    int roll_no;  
    long int phone no; };
```

Abstract Data Types

- A definition of data type solely in terms of
 - Set of related data items (or values)
 - Set of operations on the data
- Separation of logical properties from the implementation details
 - Hide implementation details (**Encapsulation!**)
- **What** not **how** is focus
- ADT tells what is to be done and data structure defines how to do it

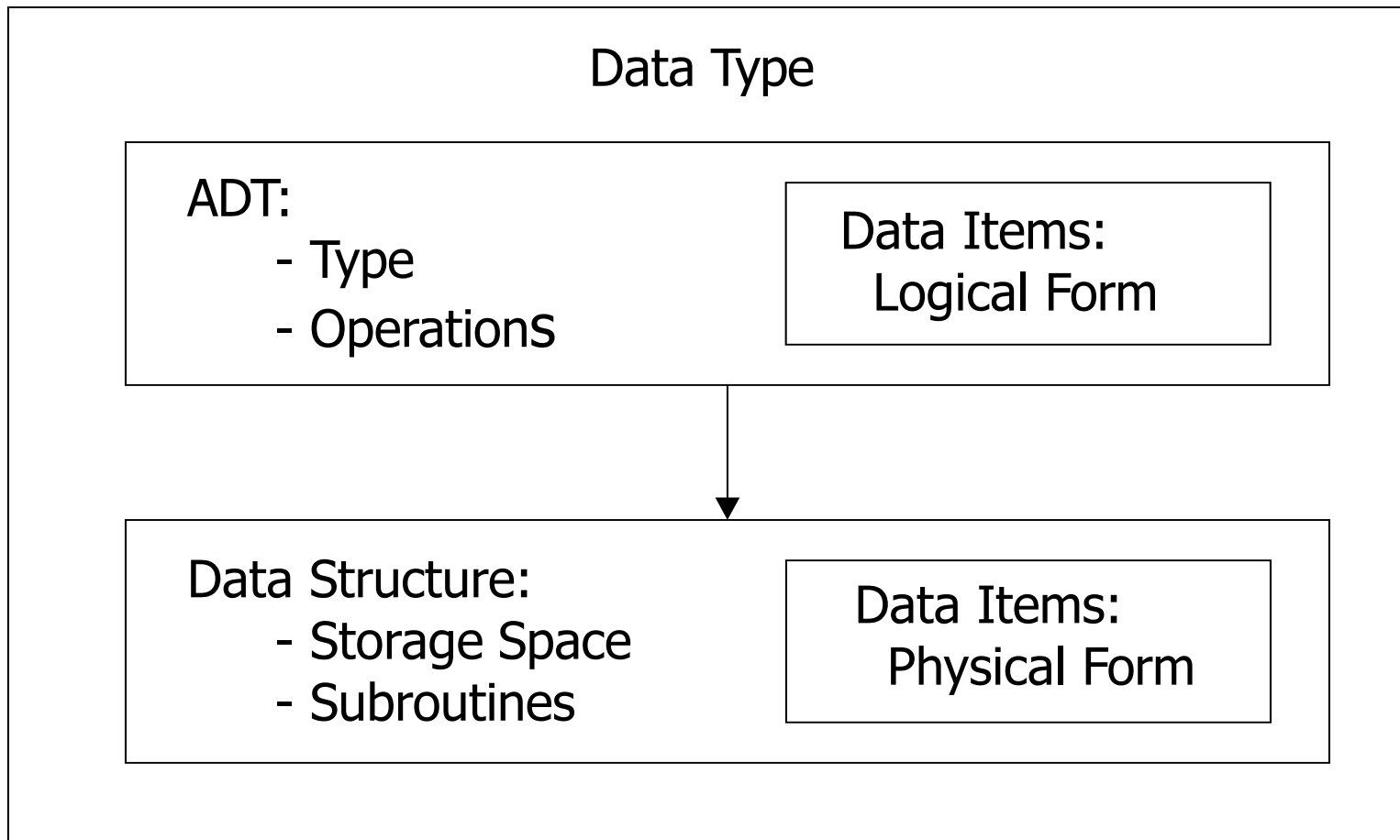
Abstract Data Types

- ADTs definition consists of blue print or logic
 - Storage structures (i.e., data structures) to store data items
 - Algorithms for basic operations
- Storage structures/data structures used in the implementation
 - Provided in a language (primitive or build-in)
 - Built from the language constructs (user-defined)
- Separation of a data type from its implementation

Data Structures

- A data structure is physical implementation of an ADT
 - Each operation associated with ADT is implemented by one or more subroutines in the implementation
- **Data structure** usually refer to an organization of data in main memory
- **File structure** is an organization of data on peripheral storage such as disk drive

ADT vs Data Structures



Example: Airplane Flight Reservation

Consider example of an airplane flight with 10 seats to be assigned?



Example: Airplane Flight Reservation

- Data
 - 10 seats
- Operations
 - List available seats
 - Reserve a seat
- Implementation: How to store, access data?



Method 1: 10 individual variables

LIST AVAILABLE SEATS:

```
if seat1 == '';  
    display 1  
  
if seat2 == '';  
    display 2  
  
.  
  
.  
  
if seat10 == '';  
    display 10
```

RESERVE A SEAT:

Set DONE to false

```
if seat1 == '';  
    ◦ print "do you want seat #1??"  
    ◦ Get answer  
    ◦ if answer=='Y';  
        ◦ set seat1 to 'X' set Done to True  
  
if seat2 =='' and Done == false;  
    ◦ print "do you want seat #2??"  
    ◦ Get answer  
    ◦ if answer=='Y';  
        ◦ set seat2 to 'X' set Done to True  
  
.  
  
.
```

Method 2: An Array of variables

List available seats:

```
for number ranging from 0 to max_seats-1, do:  
    if seat[number] == ' '  
        Display number
```

Reserve a seat:

Reading number of seat to be reserved if

```
seat[number] is equal to ' '  
    set seat[number] to 'X'
```

else

Display a message that the seat having this number is occupied

Example: Airplane Flight Reservation

- This simple example illustrate the concept of an Abstract Data Type
- ADT consists of
 - Collection of data items
 - Basic operations that must be performed on them
- In the example, a collection of data is a list of seats
- Basic operations are
 - List available seats
 - Reserve a seat

Which data structure is used for particular ADT?

- Different data structures compared for time and space efficiency.
- One best suited is selected according to current requirement of the user.
- Example: Stack
 - array → time efficient
 - linked list → space efficient
- User decide he needs time or space.

Advantages

Efficiency

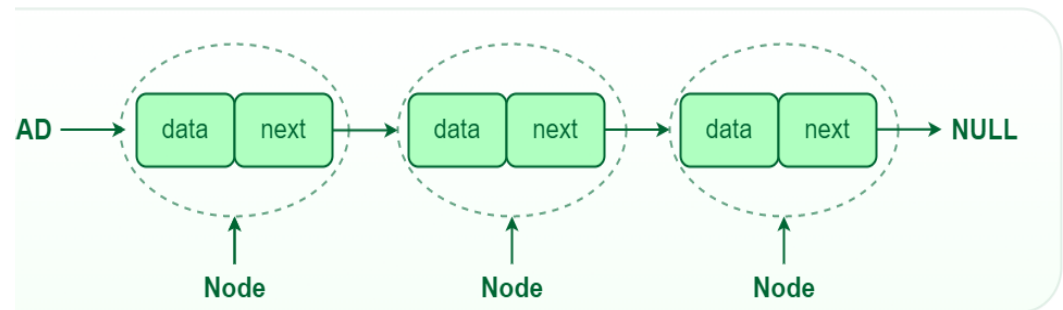
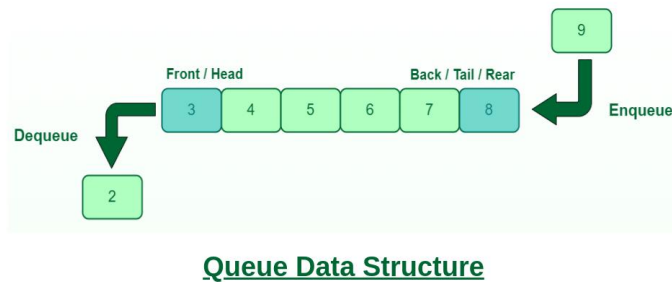
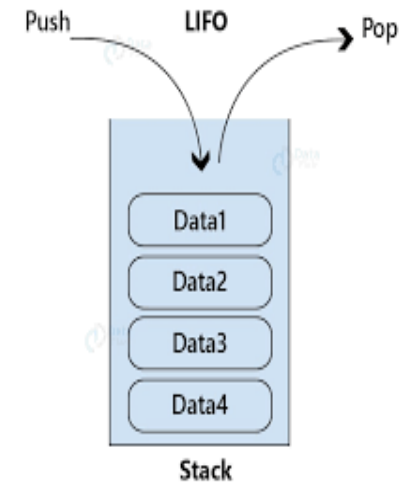
Resuability

Abstratction

Data Structures - Types

- Linear

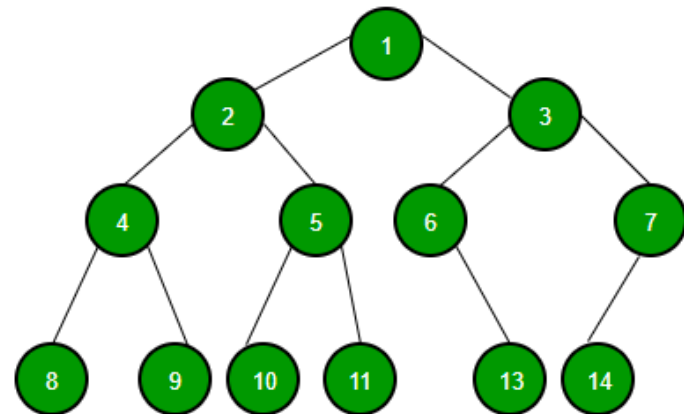
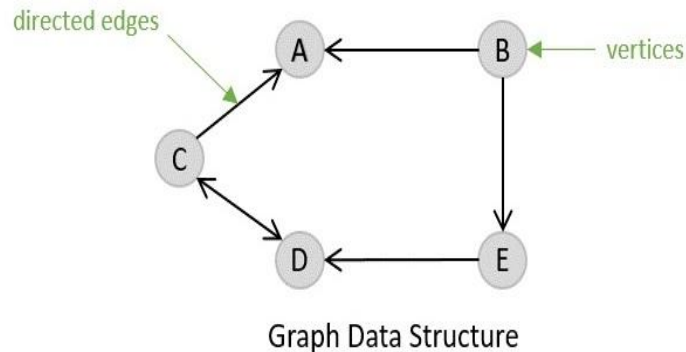
- Data arranged in linear (sequential) order.
- Example:
 - Array, Queue, Stack, Linked List
- Must have one predecessor and successor.



Data Structures - Types

- Non Linear

- Data arranged in non linear (sequential) order.
- Example:
 - Tree, Graph
- Not exactly have one predecessor and successor.



Data Structures - Types

STATIC

- Memory is allocated at compile time
- Maximum size is fixed
- Advantage:
 - fast access
- Disadvantage:
 - slower insertion and deletion
- Example:
 - Array

DYNAMIC

- Memory is allocated at run time
- Maximum size is not fixed
- Advantage:
 - fast insertion and deletion
- Disadvantage:
 - slower access
- Example:
 - Link List

Any Question So Far?

