ENSF ENSF 593/594
Data Structures - Introduction

Mohammad Moshirpour

Definitions

Data Structure: the organization of elementary data types into a larger, structured aggregate

- User to store data for an application
- May be directly supported by a programming language
  - E.g. Arrays and structs in C
- Usually created by a programmer
  - Reusable code for a data structure may be kept in a library
    - E.g. Vector class in java.util

Definitions (cont'd)

Algorithm: a well-defined set of instructions for solving a problem

- May be expressed:
  - Informally (e.g. in plain English)
  - Formally, using specially designed mathematical notations
- Is abstract
  - Is independent of its implementation (i.e. code written in a particular language)

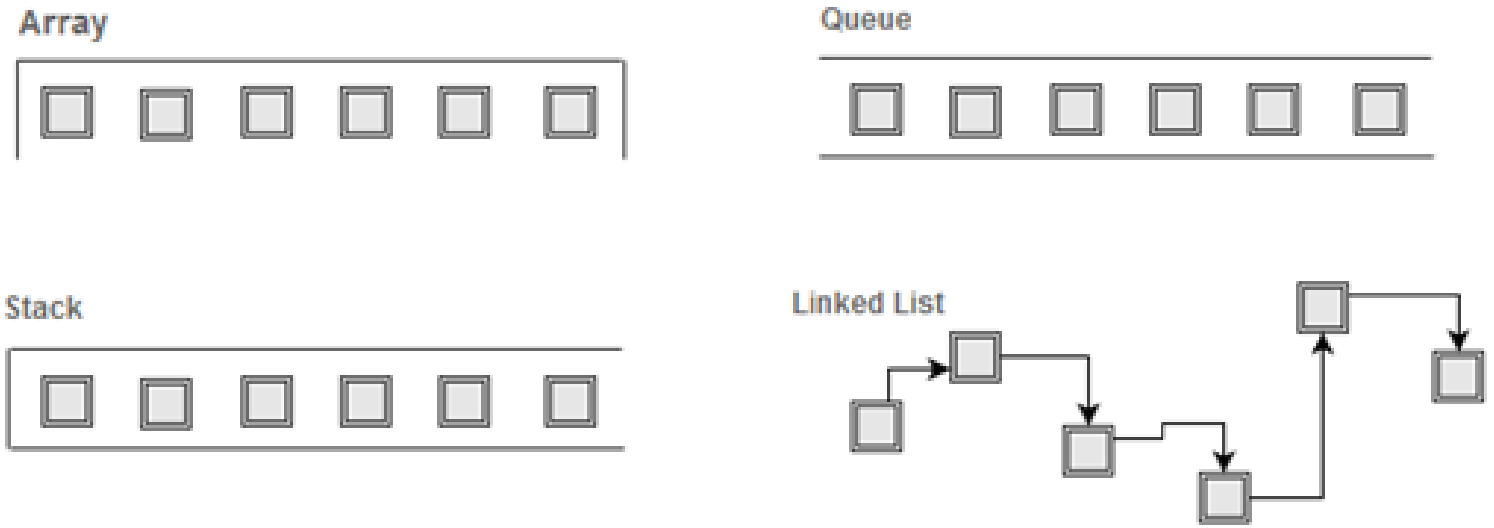
Definitions (cont'd)

Abstract Data Type (ADT): a data structure accompanied by a set of access functions

- The implementation details are concealed from client code
  - Uses information hiding
- The functions:
  - Create objects of the ADT
  - Access the contents of the data structure
- Classes in OO language are ADTs where the concealment is enforced by language syntax
- E.g. Stack ADT
  - Access functions: new, push, pop

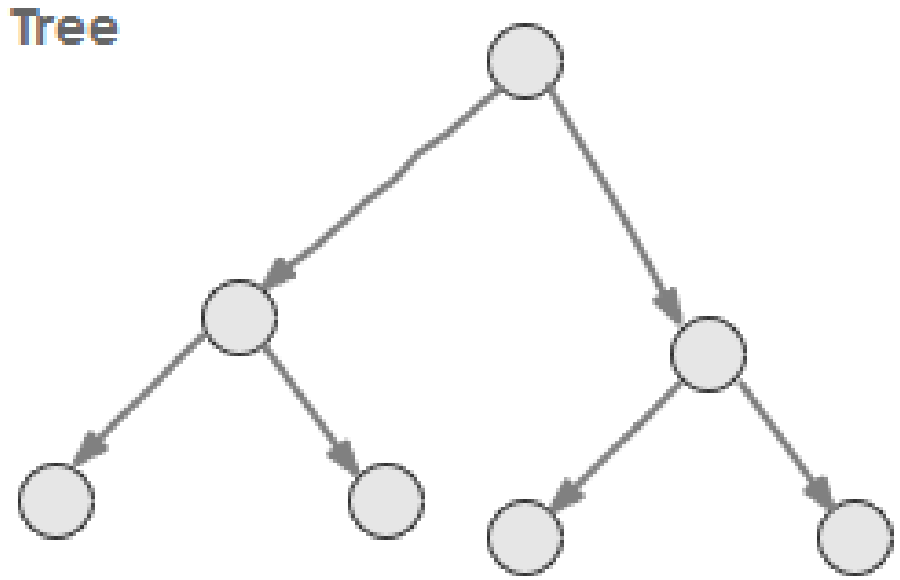
Classification of Data Structures

- Linear Structures



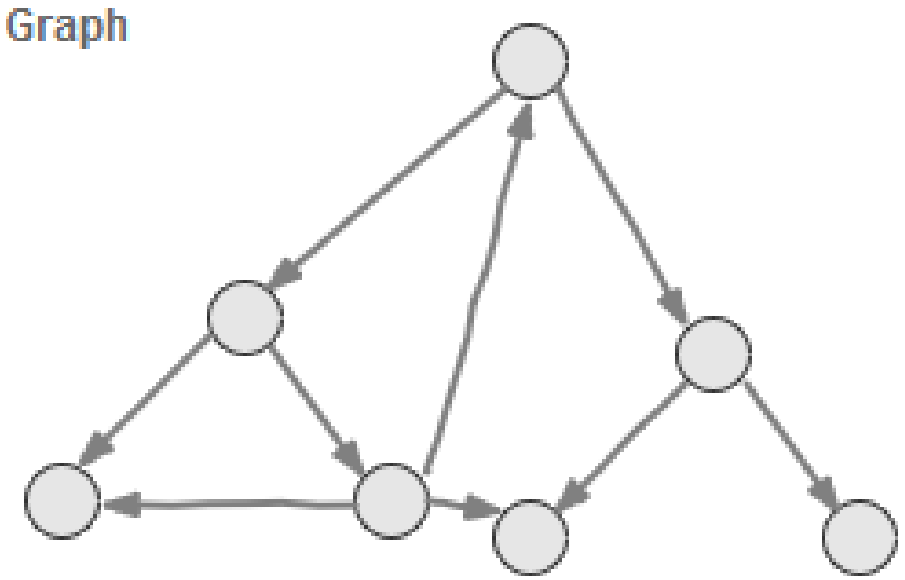
Classification of Data Structures (cont'd)

- Hierarchical Structures (Trees)



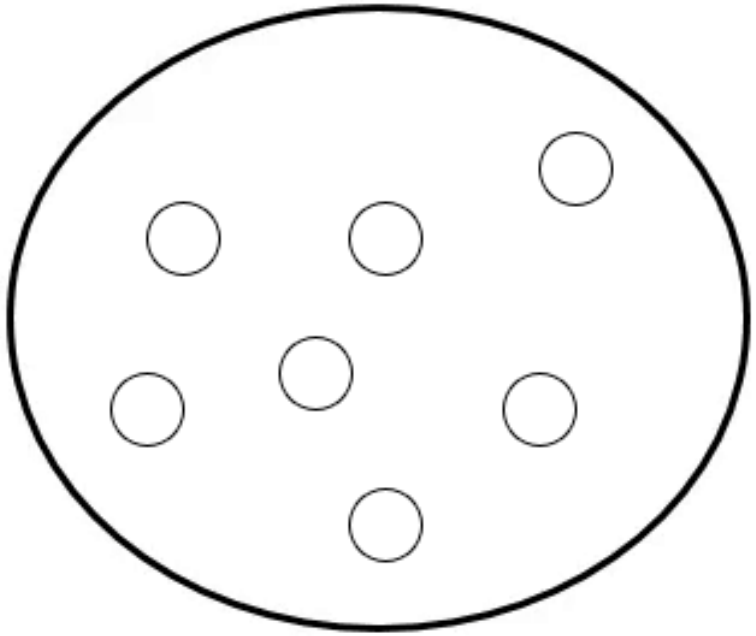
Classification of Data Structures (cont'd)

- Graph Structures



Classification of Data Structures (cont'd)

- Set Structures



Operations on Data Structures

- Most data structures are *dynamic*
  - i.e. they can grow larger and smaller over time
- *Modifying operations* change the size of the data structure
  - *Insert:* adds a record to the data structure
  - *Delete:* removes a record

Operations on Data Structures (cont'd)

- *Querying operations* return information from the data structure
  - *Search:* returns a pointer to a record that matches a key value, or nil if there is no match
  - *Minimum:* returns the record with the smallest key
  - *Maximum:* returns the record with the largest key

Operations on Data Structures (cont'd)

- *Successor:* given some records, returns the next larger record, or nil if the record is the maximum record
- *Predecessor:* given some records, returns the next small record, or nil if the record is the minimum record

Operations on Data Structures (cont'd)

- *Other operations* modify the contents of a record in the data structure
  - *Replace:* replaces an entire record with another
    - Could be done with a delete and insert
  - *Update:* overwrites one or more fields in a record

Any questions?