Faculty of
Information Technology
King Mongkut's Institute of Technology Ladkrabang

Introduction to Data Structures and Algorithms

Dr. Sirasit Lochanachit

Outline

1. Resources
2. What is a “Data Structure”?
3. What is an “Algorithm”?
4. Prerequisites
5. Topics
6. Grading

Resources

1. Course Website: https://github.com/noswolf/DSA_BIT/tree/DSAP_23
2. Google Colaboratory
    - Interactive notebooks

What is a “Data Structure” ?

How do we store, organise, and retrieve data on a computer?

What is a "Data Structure" ?

- Way to **store** and **organise** data
- Enable efficient **access** and **modification** of data
- Designed for a specific algorithm
  - Strengths and limitations
  - Time and space complexity

Abstract Data Type

- A data type where only **behavior** is defined but not implementation.
- Examples: Array, List, Map, Queue, Set, and etc.

Common vs Abstract Data Type

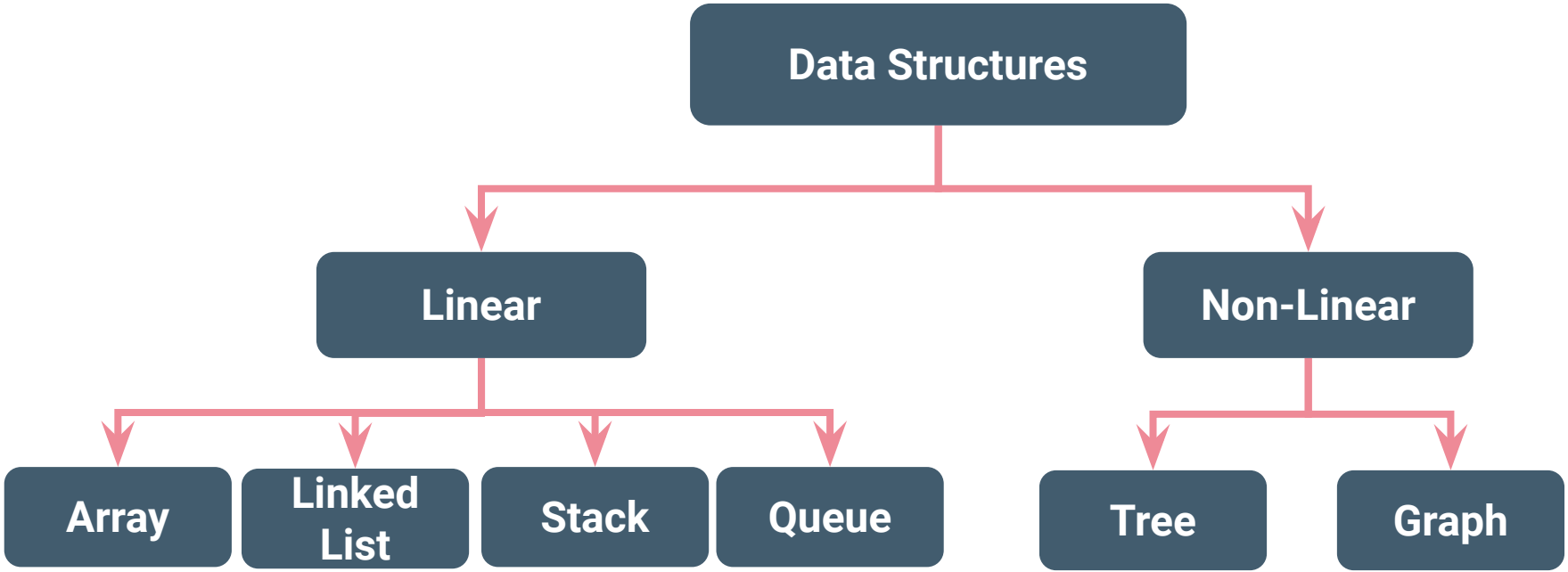
Common

- Integer
- Floating-point number
- Character
- String
- Boolean
- etc.

Abstract

- Array
- List
- Map
- Queue
- etc.

Type of Data Structure



Check out for a comprehensive list of data structures at
https://en.wikipedia.org/wiki/List_of_data_structures

What is an "Algorithm" ?

- Well-defined procedure or set of instructions to
  - transform input to output or
  - accomplish a task or
  - solve a computational problem

Why care about an “Algorithm” ?

How can we efficiently (in space/time) carry out some typical data processing operations?

How do we analyze and describe their performance?

Example: Sorting numbers

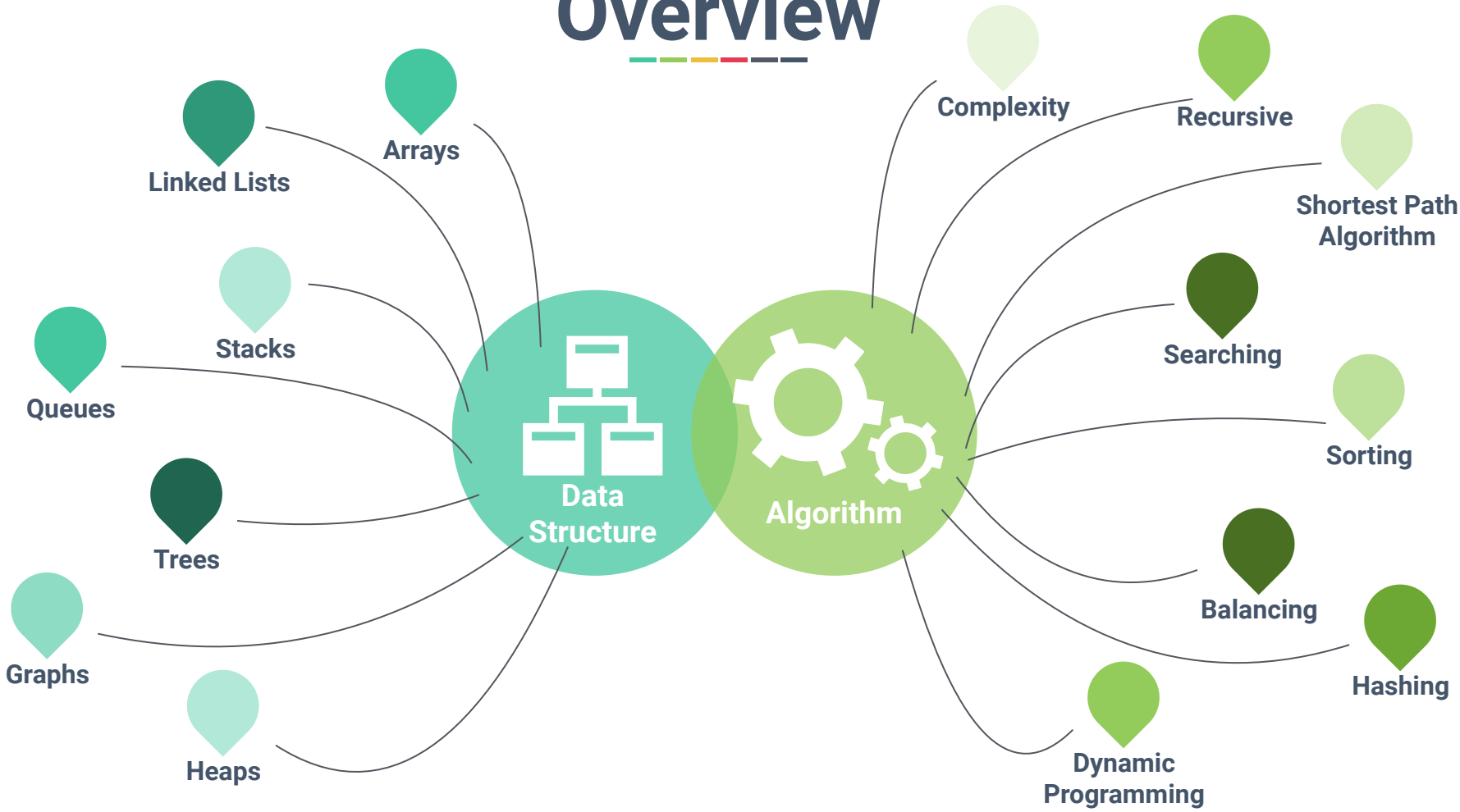
1. Input:

2. Sorting Algorithms
3. Output:

Overview

- Data Structures
- Algorithms

Overview


Linked Lists
Arrays
Stacks
Queues
Trees
Graphs
Heaps
Data Structure
Algorithm
Complexity
Recursive
Shortest Path Algorithm
Searching
Sorting
Balancing
Hashing
Dynamic Programming

Prerequisites

- Fluent in Python Programming
- Comfortable with development processes
  - Writing a function
  - Debugging and testing a code

Lesson Plan (till Midterm)

Week	Topics	Individual Assignments
03/07/2023	Python Crash Course	#1
10/07/2023	Algorithm Analysis	#2
17/07/2023	Arrays	#3
24/07/2023	Stacks [VDO]	#4
31/07/2023	Queues	#5
07/08/2023	Linked Lists	
14/08/2023	Linked Lists (Cont.) [VDO]	#6
21/08/2023	Trees	

Lesson Plan (after Midterm)

Week	Topics	Individual Assignments
04/09/2023	Search Trees	#7
11/09/2023	Search Trees (Cont.)	-
18/09/2023	Searching and Hashing	#8
25/09/2023	Sorting	#9
02/10/2023	Recursion and Sorting	-
09/10/2023	Graphs	#10
16/10/2023	Graphs (Cont.)	-
23/10/2023	-	-
	Final Exam	

Grading

Attendance	5%
Class Notes	5%
Individual Assignment	30%
Midterm Exam	30%
Final Exam	30%

Submit on MS Teams
Team Code: qm9svn3

MS Teams Link

Class Notes

For each lecture, submit your own notes either on

lecture slide

or

your notebook

Can be handwritten or typed

Submit as a **pdf file** only on MS Teams

Late Policy

Class notes: flexible over semester

- Hard deadline: 30 October 2023

Individual Assignments: Late submission is not allowed

- Hard deadline: normally 1 or 2 week(s) after posted

Reading List

Essential

Goodrich, M.T., Tamassia, R. and Goldwasser, M.H., 2013. ***Data structures and algorithms in Python***. John Wiley & Sons Ltd.

Recommended

Cormen, T.H., Leiserson, C.E., Rivest, R.L. and Stein, C., 2022. ***Introduction to algorithms***. MIT press.

Miller, B.N. and Ranum, D.L., 2011. ***Problem solving with algorithms and data structures using python***, 2nd ed. Franklin, Beedle & Associates Inc.