



بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

# ***Data Structures & Algorithm Analysis***

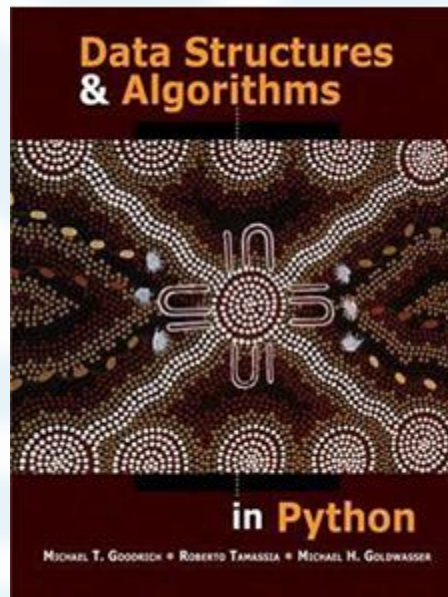
***Dr. Ali Valinejad***

**Valinejad.ir**  
**valinejad@umz.ac.ir**

**University of Mazandaran**

# Course Resources

**Textbook:** *Data Structures and Algorithms in Python*, by Goodrich, Tamassia & Goldwasser, 2013



# Grading

- ❖ *Grades are earned, not given.*
- ❖ *Judgment is based on performance, not personality.*
- ❖ *The harder you work, the more you learn, the better your performance.*
- ❖ ***Grades are weighted as follows:***
  - 10% = Projects/Programming Assignments
  - 40% = Midterm Exam (programming)
  - 15% = Final Exam (programming)
  - 35% = Final Exam on course content

# What The Course Is About

**Programming** requires **efficient algorithms** for **accessing data** in computer.

- This **efficiency** is directly linked to the **structure** of the **data** being processed.
- **Data** are simple values or sets of values.

**Program** = **Algorithms** + **Data Structures**

- **Algorithm** is a set of rules for solving a problem in a finite number of steps.
  - **Data structures** is a particular way of **storing** and **organizing data** in a computer so that it can be used efficiently.
  - **Data structures** is a systematic way of **organizing** and **accessing data**.
  - **Data structures** is concerned with the **representation** and **manipulation** of **data**.
- ❖ we are interested in the design of “*good*” **Data Structures** + **Algorithms**.

# What The Course Is About

**Program** = **Algorithms** + **Data Structures**

All **programs** **manipulate** data. So, all **programs** **represent** **data** in some way.

**Data manipulation** requires an **algorithm**.

We shall study **ways to represent data** and **algorithms to manipulate** these **representations**.

# Outline

- Complexity
- Recursion
- Array-Based Sequences
- Stacks, Queues, and Deques
- Linked Lists
- Trees
- Priority Queues
- Maps, Hash Tables, and Skip Lists
- Search Trees
- Sorting and Selection
- Graph Algorithms