International University

# Data Structures and Algorithms

Tran Thanh Tung

# What are they?

- What is a data structure?
- What is an algorithm?
- Not yet to answer but let's see examples:

# Examples

- You have a book-shelf.
  - How to arrange books into the book-shelf?
  - Why ?
- To travel to all famous tourist sights in HCM city.
  - What route and schedule should you follow?
  - Why should you do like that?

# What are they?

A data structure is

- an arrangement of data
- in a computer's memory (or sometimes on a disk).

# What are they? Algorithms

Are sequences of instructions

- To manipulate the data in these data structures
- in a variety of ways.

Such as

- Insert a new data item
- Delete a specified item
- Iterate through all the items in a data structure
- Sorting in-order all items in a data structure

# Who are you? And why?

- Familiar with OOP in Java/.NET
- Basic programming skill in Java/.NET
- Why you learn this course?
  - DSA is one of the most fundamental course in CS and IT.
  - Provide necessary knowledge to learn further: Database, Operating System,….
  - Program = Data Structure + Algorithms

# Objectives

- Understand general concepts of analyzing algorithms.
- Can use basic data structures to solve practical problems.
- Know to decide which data structures and/or algorithms should be used in practical problems.
- All of them in Object Oriented Programming (OOP)

# Content of the course

- Review of OOP and Java
- Basic data structures:
  - Arrays
  - Queue
  - Stack
  - List
  - Tree
- Advanced data structures:
  - Advanced Tree
  - Graph
- Algorithms:
  - Searching
  - Sorting
  - Graph Algorithms

# Week by week topics (*)

1. OOP and Java
2. Arrays
3. Sorting
4. Queue, Stack
5. List
6. Recursion

Mid-Term

7. Advanced Sorting
8. Binary Tree
9. Hash Table
10. Graphs
11. Graphs Adv.

Final-Exam

8 LABS

# Class rules

- Attendance: on-time (MUST)
- Mobile devices: off
- Private discussion: no
- Internet search/chat: no (screen turn-off)
- Exams/tests/exercises: no make-up (unless special cases)

# References


- Class notes
- Robert Lafore, "Data structures and Algorithms in Java", Waite Group Press, 2002.
- Introduction to Algorithms [Hardcover]
  - Thomas H. Cormen, et al. The MIT Press
- Download from book publisher
  - Workshop Applets
  - Example Programs
- www.samspublishing.com
  - Search ISBN: 0672324539

# Grading policy

- Assignments + Labs : **30%**
- Mid-term exam : **30%**
- Final exam : **40%**
- You failed if Final_Score < 50!

# Projects (3-4 devs)

1. Game Minesweeper
   - With undo feature
2. Game Lines
   - With moving path
3. Game Battleship
   - Human vs Computer (with non-random move)
4. Interactive chart
   - Zoom in/out
   - Multiple functions

**Tasks**

- Read and present the game/chart rules
- Design classes
- Implement the game/chart with basic rules (50pts)
- Write report (10pts) – game/chart rules, class diagrams, …
- Demonstration (10tps)

- **Mandatory**

- Use Git (10pts) – Commits history
- Graphical User Interface (10pts)

- **Bonus**

- Extra features **(+2pts for each)**
- Applying design patterns **(+5pts for each)**
- Using C#/Javascript (**5pts**)

# Assessment

- DON'T COPY and DON'T ALLOW ANYONE COPY YOURS
  - Zero for all
- Except as otherwise noted, all assignments:
  - Are to be done solo (by yourself). As before,
    - You may discuss the assignments with other students
    - You may help (and get help with) debugging
    - You may not give your source code to anyone
- Late assignments will lose 10 points per day, and are not accepted if more than a week late

# Contact


- Tran Thanh Tung
- Email: tttung@hcmiu.edu.vn
- Office: 610