



Vietnam National University of HCMC  
International University  
School of Computer Science and Engineering



# Data Structures and Algorithms

## ★ Overview ★

Dr Vi Chi Thanh - [vcthanh@hcmiu.edu.vn](mailto:vcthanh@hcmiu.edu.vn)

<https://vichithanh.github.io>



SCAN ME

# Instructor introduction



Vi Chi Thanh



Email: [vc Thanh@hcmiu.edu.vn](mailto:vc Thanh@hcmiu.edu.vn)



Website: <https://vichithanh.github.io>



SCAN ME

# Classroom Etiquette

- + All laptop computers, cell phones, tablet computers **must be closed** during all classroom hours
- + If you wish to use a computer, you are welcome to step outside
- + Computers distract the most people behind and around the user
- + You require a **Verification of Illness** form to use a computer in class
- + The classroom is not for playing games or watching the next football match – even if your country is playing – but you are welcome to sit outside
- + Last but not least: **I have a noise problem!**



# What you have learnt...

- + The history of computing / objects / types / console I/O
- + Operators / loops / methods / parameter passing
- + Selection statements / arrays / strings
- + Exceptions / debugging
- + File input / file output
- + Pointers / unsafe code / linked lists
- + Collections / multi-dimensional arrays / search algorithms
- + Sorting algorithms
- + Object-oriented design / polymorphism / interfaces / inheritance
- + Abstract class

# What you have learnt...

- + These were the basics of programming
  - + The ability to manipulate the computer to perform the required tasks
- + You saw data storage techniques:
  - + Arrays, and
  - + Linked lists (collections were discussed)
- + You saw array accessing/manipulation techniques:
  - + Searching, and
  - + Sorting

# Algorithms and Data Structures

- + In this course, we will look at:
  - + *Algorithms* for solving problems efficiently
  - + *Data structures* for efficiently storing, accessing, and modifying data
- + We will see that all data structures have trade-offs
  - + There is ***no ultimate*** data structure...
  - + The choice depends on our requirements

# Week by week topics (\*)

1. Overview, DSA, 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**

**10 LABS**

# Classroom Etiquette

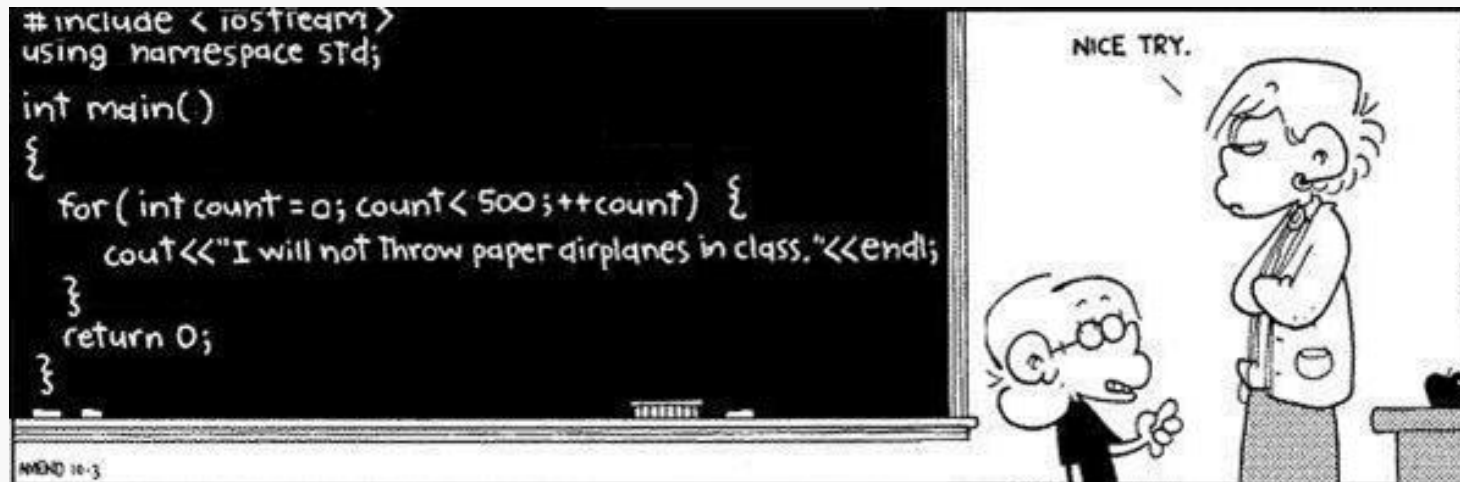
- + All laptop computers, cell phones, tablet computers **must be closed** during all classroom hours
- + If you wish to use a computer, you are welcome to step outside
- + Computers distract the most people behind and around the user
- + You require a **Verification of Illness** form to use a computer in class
- + The classroom is not for playing games or watching the next football match – even if your country is playing – but you are welcome to sit outside
- + Last but not least: **I have a noise problem!**





# Java

+ You will be using Java programming language for this course



# Java

- + This course does not teach Java programming
  - + You will use Java to demonstrate your knowledge in this course
- + One lecture covers:
  - + Features of Java and differences with procedural programming languages.
- + An on-line tutorial is available online (use Google)
  - + It assumes minimal knowledge of programming
- + Please note: marks in OOP are not strongly correlated with marks in DSA

# Java

- + Other sources of help in Java are:
  - + The Project T.A.s,
  - + The lab instructor, and
  - + The instructor
- + Laboratories are held every week, starting a few weeks later.
  - + Laboratories is associated with a project
- + IDE: You are welcome to use your preferred one.

# Evaluation

**The course is divided into numerous topics**

- + Storing ordered and sorted objects
- + Storing an arbitrary collection of data
- + Sorting objects
- + Graphs
- + Algorithm Design Techniques

# Evaluation

**Your evaluation in this course is based on three components:**

- + One mid-term examination (30%)
- + One final examination (40%)
- + Labs (30%)
  - + 8 lab assignments (30%)
  - + 1 practice examination (35%)
  - + 1 course project (35%)
  - + In-class contributions (bonus)

# Textbooks

- +Class notes
- +Lafore, R. (2017). Data Structures and Algorithms in Java. United Kingdom: Pearson Education.
- +Goldwasser, M. H., Goodrich, M. T., Tamassia, R. (2014). Data Structures and Algorithms in Java. United Kingdom: Wiley.
- +Leiserson, C. E., Cormen, T. H., Rivest, R. L., Stein, C. (2009). Introduction to Algorithms. India: MIT Press.

# Academic Offences

Academic Offences include, but are not limited to:

- + **Infringing unreasonably** on the work of other members
  - +E.g., disrupting classes
- + **Cheating**
- + **Plagiarism**
- + **Misrepresentations**

# Plagiarism

## **All projects must be done individually:**

- + You may not copy code directly from any other source
- + Plagiarism detection software will be used on all of the projects
- + If you viewed another code (from books or lecture notes), you must include a reference in your project
- + You may not share code with any other students by transmitting completed functions to your peers
  - + This restriction includes - but is not limited to - electronic and hard-copy sharing
- + You may discuss projects together and help another student debug his or her code; however, you cannot dictate or give the exact solution



# Plagiarism

- + Collaboration with other students must be limited to
  - + Discussions
  - + High-level pseudocode
  - + Assistance with debugging (only through the offering of advice)
  - + Sharing test files
- + All such collaborations **must** be documented in your source code

# Plagiarism

- + When one student copies from another student, both students are responsible
  - + Exceptions are made for outright theft
- + The penalty for plagiarism on a Project is a mark of 0 on the project in question and a further 5% is subtracted from your final grade
  - + Regardless if Projects are counted or not

# Plagiarism

- + One student cannot accept “full responsibility”
  - + For example, Alex, Bailey, Casey, and Devin worked together in a group
  - + They each did their own work. However, they shared code to comment on each others programming
  - + Bailey gave Alex’s code to Emerson who copied it for his project and submitted it
  - + Alex, Bailey and Emerson received a 0 and -5 %

# Plagiarism: Example 1

- + Alex and Bailey were lab partners
- + Bailey left herself logged on to her account to allow Alex to complete the lab
- + Alex copied Bailey's project

# Plagiarism: Example 2

- + Leslie asked if Morgan could send her his code so that she could look at it (promising, of course, not to copy it)
- + Morgan sent the code
- + Leslie copied it and handed it in

# Plagiarism: Example 3

- + Erin did not change her default password
- + Fanny logged onto Erin's account and took Erin's code
  - + Erin is still responsible

# Plagiarism: Example 4

- + Garry and Harry worked together on a single source file initially and then worked separately to finish off the details
- + The result was still noticeably similar with finger-print-like characteristics which left no doubt that some of the code had a common source

# Plagiarism: Example 5

- + Jordan uploaded the projects to GITHUB.com without setting appropriate permissions. Kasey found this site, downloaded the projects and submitted them. Both are guilty.
- + This applies to any public forum, news group, *etc.*, not just github.com...



# Plagiarism

- + The minimum penalty for plagiarism is 0 on the project and -5% on your final mark for each case of plagiarism
  - + The penalty is applied regardless of what proportion the Projects are of your final grade
- + A student who cheats must receive a grade lower than a student who did not hand in a project

# Plagiarism

+The best way to avoid plagiarism is:

- +review the Java tutorial
- +read the project as soon as it is available
- +start the project so that there is sufficient time to contact the T.A. or myself if you have difficulty
- +do not give your code to anyone

# Distribution of Information

Information may be pass to the class through one of two media:

- An announcement in class,

- An announcement on Blackboard

# 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).

+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 within 24 hours will lose 25%. Submission later than 24 hours will NOT be accepted.

# Projects (group of 2)

## (example projects below)

### 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 rules
- + Design classes
- + Implement the game with basic rules (50pts)
- + Write report (10pts) - game rules, class diagrams, ...
- + Demonstration (10pts)

### + 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)



Vietnam National University of HCMC  
International University  
School of Computer Science and Engineering



**THANK YOU**

Dr Vi Chi Thanh - [vcthanh@hcmiu.edu.vn](mailto:vcthanh@hcmiu.edu.vn)

<https://vichithanh.github.io>



SCAN ME