Problem Solving

Course Introduction

Instructor: Jae-Pil Heo (허재필)

About the Instructor

- Jae-Pil Heo (허재필)
- Joined SKKU in 2017
- Visual Computing Lab @ SKKU
  - Mainly focuses on Computer Vision and Machine Learning
  - 7 master, 7 undergraduate students + me

- E-mail: jaepilheo@skku.edu
- Office: 85472 (4th floor in Corporate Collaboration Bd.)
- Phone: 031-290-7965

Acknowledgements

- The materials are built upon previous efforts of:
  - Prof. Kyung-Yong Chwa (KAIST)
  - Prof. Chang Wook Ahn (GIST)
  - Prof. Jinkyu Lee (SKKU)

Course Information

- Mon 16:30 – 17:45, Wed 15:00 – 16:15
- 3 credits
- Language: English

- This course is designed for the second-year students

- Prerequisites
  - High-school math (matrix, vector, …)
  - Data structures (tree, graph, …)
  - C programming language
  - If you are unsure, consult the instructor at the end of this class.

Course Objectives

- Problem solving techniques for computer science and engineering
  - Deal with the problems creatively and effectively
  - Develop computational thinking
  - Implement ideas using C programming language
  - Present your idea to your classmates
  - (Know your departmental students and make friends)

- This course is designed for those who have not taken the Algorithm course.

- Problem-based course: *It seems unorganized!*

Computational Thinking

- Computational thinking [Wikipedia]
  - Problem formulation (abstraction)
  - Solution expression (automation)
  - Solution execution and evaluation (analysis)

- + Implementation
- + Presentation
- + Discussion

Overview

- Lecture
  - I will give you a few lectures about problem solving strategies and complexity analysis

- Problem Solving
  - You will solve 1 or 2 problems for each week
  - Present and discuss your solutions in class
  - **Your participation is very important!**

Grading

- Homework: 80%
  - (Mostly) You are asked to solve problems and submit a report
  - A few programming assignments (in C)
  - Late submission is not allowed

- Group projects: 20%

- Extra credits for in-class presentation
  - +1 ~ +3 points for each presentation
    - The same scale with homework. Each homework problem = 10 pts
  - +0.x points for good questions or feedbacks
    - State your major and name when you give questions or feedback to the presenter

- No exam ☺

Homework Evaluation Protocol

- Correctness of your solution
  - Check whether your algorithm has counter-examples.
- Efficiency of your solution
  - Time and space complexities
- Clarity of your explanation
  - TAs are not a compiler. Avoid to write code if possible.
  - Describe your solution first, and give an example

- Additional comments:
  - Do not write in cursive letters
  - Do not discuss too much with your friends or google.
  - Spend enough time to develop solution.

Class Attendance Rule

- The following cases will be regarded as presence:
  - Department/college activities
  - Military services (e.g., 예비군훈련)
  - Your family passed away; go there immediately and let me know later

- Every two absences → lower your grade (e.g., A+ → A)
  - No credit for any lateness
  - Exemption for the first absence
  - Example
    - late + absence 3 – 4 times → 1 lower grade
    - late + absence 5 – 6 times → 2 lower grade
    - late + absence 7 – 8 times → 3 lower grade
    - ….. → F

Official Language in Class

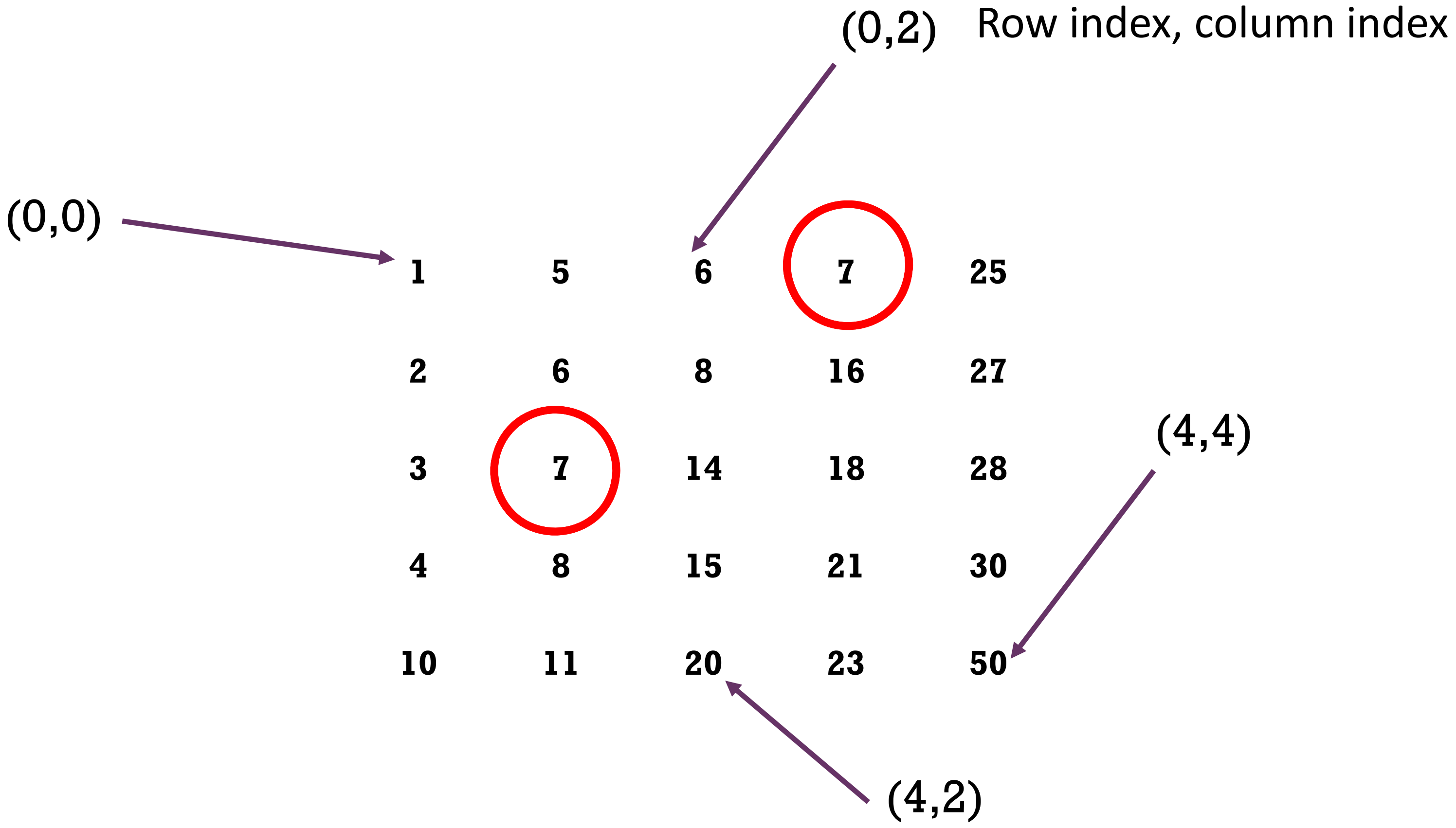
- Lecture
  - I’ll give lectures in English
  - I may explain again in Korean if materials are unclear to you
- Presentation/discussion/questions in class
  - You’re highly recommended to use English.
- Assignments
  - Comments in source codes: English
  - Your report: either in English or Korean
- Exam
  - Questions will be given in English.
  - Your answers: either in English or Korean

Sample Problem #1: Finding k

- Finding k
  - There is an $N \times N$ matrix. All elements in each row and column are sorted in an ascending order.
  - Find a single k efficiently (in terms of number of element accesses). If there are multiples k in the matrix, you need to find ANY SINGLE k.

Sample Problem #1: Finding k

- Example
  - Input: 5×5 matrix, $k = 7$
  - Output: (2,1) or (0,3)



(0,2) Row index, column index

(0,0)

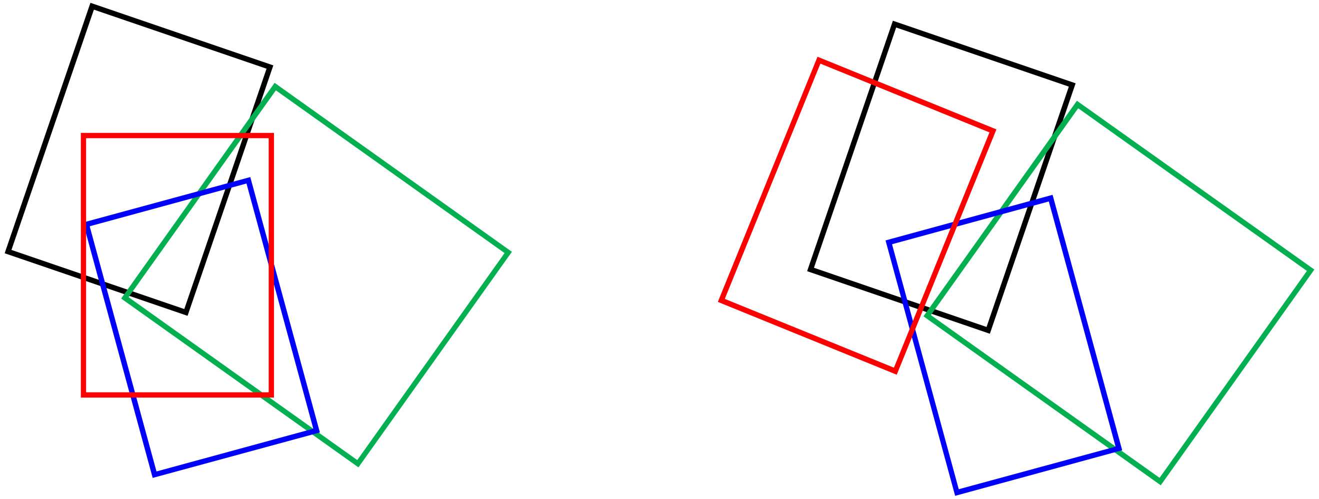
1	5	6	7	25
2	6	8	16	27
3	7	14	18	28
4	8	15	21	30
10	11	20	23	50

(4,4)

(4,2)

Sample Problem #2: Papers

- Papers
  - We have N rectangular papers on a desk.
  - We want to pin them down with a pin.
  - Given positions, orientations, and sizes of papers.
  - Design an algorithm that determines whether they can be pinned with only one pin or not.
- Example:



Group Project #1: Human Sorting

- Each member will have 15 cards.
- An integer between 100 and 999 is written on each card.
  - There can be multiple cards with the same number.
  - Each team will prepare a deck of cards. A deck for playing will be randomly chosen.
- The goal is to sort all the cards while minimizing time.
  - The evaluation is mostly based on the sorting time
- Each team will play twice and be evaluated by the average sorting time.

Project #1

- There is only a single physical operation that each member can do with other members:
  - A member can give some cards to another team member ***beside*** him/her.
  - You can use a linear or circular formation.
  - You can communicate

- During sorting, all cards should be in your hands.
  - Each player should be holding at least 1 card in hands.

- Finally, each member should have 15 cards
  - The team leader should declare the end of your play.
  - Collected cards from the last member to the forefront member should be sorted.

Any Question?