HANDONG GLOBAL UNIVERSITY

Syllabus

Algorithm Analysis

School of CSEE

Course info. overview

◆ Instructor : Dr. Whanki Yong

Office : NTH 307

O.H. : Thursday. 14:30~15:30, email

Phone : 260-1375

Email : wyong@handong.edu

◆ Classes : LMS based

Mon.: mainly LMS

Thur.: In classroom (Flipped learning)

◆ T.A.

이준명(218*****), 한성화(218*****), 서주은(220*****)

Lecture Overview

Algorithm is a strategy for solving a problem with the assistance of a computer. To say that a problem is solvable algorithmically means, informally, that a computer program can be written that will produce the correct answer for any input if we let it run long enough and allow it as much as storage space as it needs. In this course, we will study various algorithms by analyzing them and discuss various algorithm design techniques. C++ will be used to develop and implement programming assignment.

- Text book :

  "*Introduction to Algorithms*", Cormen, Leiserson, Livest, and Stein, Any edition, The MIT press.

- Prerequisite :

  Data Structure

Programming language

◆ Compiler

Official one: g++ // editor: VsCode

◆ C++ 강의

1. KOCW: 금오공대 임기무교수
2. KOCW: 건국대학교 지정희교수

Grading Policy

- Quiz (2) : 15 points
- Homeworks : 35 points
- Midterm : 25 points
- Final : 25 points

◆ Above policy may be subject to change with prior notice.

◆ Exam: In classroom

◆ Attendance: Handong rule

Homework

@ Note :

- Written homework : No late homework is allowed. Homework should be handed in time. Late homework will not be accepted.

- Programming homework : There will be 20% deduction for the first day, and additional 10% deduction for the following days.

  1 day late : 20% deduction 2 day late : 30% deduction

  3 day late : 40% deduction...

  * Sunday and Holidays are not counted.

- Remember Handong Honor Code!!! Plagiarism will not be tolerated!!!

- Also, there will be no make-up quiz/exam under normal circumstances.

honor code guidelines

- Any type of dishonesties will result in failure (F).

- For the criteria of dishonesty, see HGU CSEE Standard
  - Korean version: https://drive.google.com/file/d/0B9iQGS7v1k9ORGhXSHNyTkpvQW8/view?usp=sharing
  - English version: https://drive.google.com/file/d/0B9iQGS7v1k9Ob0oxTExmMjhPU28/view?usp=sharing

- In HWs, you can discuss and share your ideas, but do **NOT** show your code (implementations).
- Do **NOT** copy any implementation from the web.
- Also, when you get an idea from web. or any materials, you have to specify it. (use comment)
- 족보?

Tentative Course Schedule

Weeks	Lecture Topics
1	Introduction / Analyzing & Designing Algorithms
2	Asymptotic Notation / Recursion & Induction
3	Exercise / Dynamic programming
4	MCM problem / LCS problem
5	Greedy algorithms / Quiz1
6	Quiz1 review / Huffman code
7	Knapsack problem / Graph
8	Graph / Midterm

Tentative Course Schedule

Weeks	Lecture Topics
9	Midterm review / Topological sort / Critical path / Biconnected
10	MST(e-learning) / Bellman Ford / DAG
11	Dijkstra, Floyd algorithm / Rabin-Karp algorithm
12	KMP algorithm / Red-Black tree
13	Average case analysis / Quiz2
14	Quiz2 review / Linear time sorting
15	Order statistics / NP-completeness / Approximation algorithm
16	Final week

Note

- Beginning on 2nd week, it will be flipped learning

  (Mon.: LMS, Thur.: practice prob. /quiz / exam)
- No zoom class under normal circumstances.
- You may use classroom even in zoom class, if there is any..
- No cell. Phone, no SNS, etc.
- Do your best to concentrate on the lecture.
- Online class recording

  - Zoom recording will not be provided.

  - For officially approved absence: zoom recording may be provided

Note

- You may record lecture (audio only) and the recorded lecture should not be used other purpose than studying. Also you should get a permission before you record the lecture.
- Be a professional.
- Ask T.A. for attendance and scores.
- Ask instructor for exam, class, etc. (문의 게시판)
- Scores should be corrected, but grade never be negotiated.
- Read textbook.

HANDONG GLOBAL UNIVERSITY

Introduction

Algorithm Analysis

School of CSEE

Problems and Techniques

- Problems

  Ex) Sorting

  Selection (minimum, maximum, *k*-th smallest element)

- Techniques - Strategy

  - Divide-and-conquer
  - Backtracking
  - Greedy
  - Dynamic Programming

Data Structure

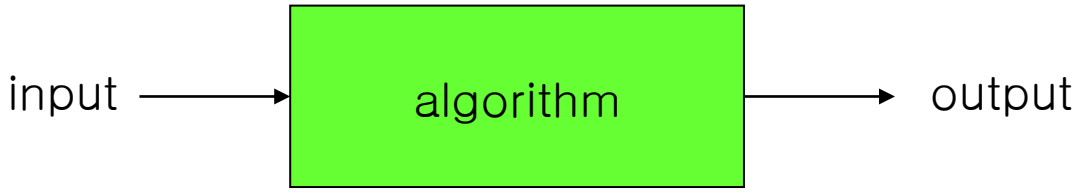
- Data structure plays an important role when solving a problem.

  Ex) For a search problem,

  - Linked list
  - Binary search trees
  - Hash tables
  - Linear array

What is Algorithm?

- Definition: A well-defined computational procedure that transforms input – some value, or set of values - into a desired output.



Efficient in terms of execution time

Statement of the problem specifies the desired input/output relationship. Algorithm is a step-by-step description of a procedure which, if followed closely, produces a well-defined result.

What is Algorithm?

- Example : sorting problem
  - input : a sequence of *n* numbers <*a*1,*a*2,…,*a*n>.
  - Output : a permutation (reordering) <*a*1', *a*2',…,*a*n'> of the input sequence s.t. *a*1' ≤ *a*2' ≤ … ≤ *a*n'.
- *Instance* of a problem: a particular input of the problem

  Example) Sort <31,41,59,26,41,58>
  - Input sequence <31,41,59,26,41,58> is an instance of the sorting problem.
  - Output : sequence <26, 31, 41, 41, 58, 59>

An Example: Insertion Sort

```
InsertionSort(A, n)
  for i = 2 to n {
     key = A[i]
     j = i - 1;
     while (j > 0) and (A[j] > key) {
          A[j+1] = A[j]
          j = j - 1
     }
     A[j+1] = key
  }
```

Expressing algorithms

- We express algorithms in whatever way in the clearest and most concise.
- Often just use English.
- To make clear issues of control, use pseudocode – part code, part English - similar to programming language.

Correctness & Efficiency

- An algorithm is said to be ***correct*** if, for every input instance, it halts with the correct output.
- Even if computers are very fast and memory is very cheap, you would still want your implementation to be within the bounds of good software engineering practice.