Algorithm Design & Analysis (CSE222)

Lecture-1

Outline

- General info & Policies
- Course Outline
- Introduction
  - What are algorithms
  - Examples

General info & Policies

Lecture Days: Monday and Wednesday @C201

Time: 2pm to 330pm

Tutorials: Tuesdays 1:30pm to 3pm. (Tomorrow no tutorials)

General info & Policies

Instructor: Supratim Shit

Office Hours: By appointment, Tuesday 10-12 at B512, R & D Block

Always send calendar invite (to supratim@...) and book your appointment.

Teaching Fellow (TF): To be announced.

Teaching Assistants (TA): To be announced.
Every students will be assigned to one TA for doubt clarification.

General info & Policies

Addressing queries/doubts:

Your Friends/Peers ⇒ Your TA ⇒ Other TAs ⇒ TF ⇒ me.

General info & Policies

Course Page: bit.ly/ADA-b

Class code: ubidmle

All announcements will be posted in google classroom.

All evaluation components are to be submitted and graded in this google classroom.

Do not post your solutions in public.

General info & Policies

Evaluation Scheme

- Attendance: 5%
- Quiz: 15% (n-1)
- Assignments:
  - Theory: 15% (Only pdf submissions are allowed. Latex template will be shared soon)
  - Programming: 10% (Platform to be announced soon. Programming language should be C++/Java, not Python)
- Mid Sem: 25%
- End Sem: 30%

General info & Policies

Evaluation Type
Quizzes: Short and surprise (n-1)
Mid Sem & End Sem: Proctored offline

No Re-Quiz at any ground.

Make-up exam only for Mid-sem and End-sem with approved medical leave.

Note: Questions need not be same for both sections.

Assignments

- Theory
  - You can submit it in a group of 2 students. Mention your group members and make sure to only submit one assignment file. You may discuss the problem with your friends (other than group member) but write it in your own words and mention your collaborators name. If you have referred to online sources then mention the URL.
- Programming
  - Submit your assignments individually.

Anti-Plagiarism

- Assignment
  - First Instance: 0 in the assignment & one letter grade less.
  - Next Instance: Fail (F-grade)

- Quizzes/Exams
  - First Instance: Fail (F-grade)

Late Submission

- Assignments
  - Theory: 1 day late ⇒ 25% deduction. 2 days late ⇒ 50% deduction. 3 or more days ⇒ 100%.
  - Programming: 1 day late ⇒ 100% deduction.
- General: Always submit your assignments at least an hour before the deadline. Do not submit in the last hour of the deadline. Any GC system related issues due to last hour submission will not be considered.
- Only pdf format either from MS word, G-doc or Latex will be accepted.

General Decoram

Be polite while communicating with others.

Keep checking google classrooms for updates

Do not spam in the google classroom.

In case of discrepancy, be discreet with your arguments. Vague or irresponsible arguments will not be entertained and it can lead to reevaluation and subsequently reduction in marks.

If attendance falls below 40% then faculty may opt, not to put up the scribble in the google classroom.

Outline

- General info & Policies
- Course Outline
- Introduction
  - What are algorithms
  - Examples

Course Outline

Introduction, Recurrence relations

Divide & Conquer

Dynamic Programming

Greedy Methods

Graph Algorithms: DFS and its applications, shortest path, spanning tree

Network Flow

NP-Completeness

Outline

- General info & Policies
- Course Outline
- Introduction
  - What are algorithms
  - Examples

Algorithms

Step of process needs to carried out to solve a problem.

Sequence of instructions that acts on an input (or values) to produce an output in finite number of steps.

Input: May or may not take input. If it takes input then it must be well defined.
Output: Must produce at least one well defined output.
Finiteness: An algorithm must terminate after finite number of steps.
Definiteness: The instructions must be unambiguous. At every step there has to be one and only possible outcome.
Effectiveness: One should be able to follow each and every instructions to solve the problem.

Algorithms

Mathematics: Solving a system of linear equations

Artificial Intelligence: Used in developing intelligent systems image recognition, voice recognition etc.

Data Science: Analyze and extract insights from large datasets.

Examples

Where have you seen algorithms?

Coming to class

Finding a book in the library

Following recipes

First Algorithm

Finding largest and the smallest element in the following array A.

A =

3	8	2	6	9	4	5
A[1]	A[2]	A[3]	A[4]	A[5]	A[6]	A[7]

Let **Largest** = **Smallest** = **A[1]**

Go over every element in A and compare each with **Largest** and **Smallest**.
Update **Largest** if an item is bigger than **Largest**.
Update **Smallest** if an item is smaller than **Smallest**.

Is this correct? Why?

Pseudo Code - Find Max and Min

Input: An array **A**, Size(**A**)
Output: **Largest**, **Smallest**

```
FIND-MINMAX(A, n)
1  i = 1
2  Largest = Smallest = A[i]
3  i = i + 1
4  while i < n
5        if A[i] > Largest
6              Largest = A[i]
7        if A[i] < Smallest
8              Smallest = A[i]
9        i = i + 1
```

Find Max and Min

Input: An array **A**, Size(**A**)
Output: **Largest**, **Smallest**

```
FIND-MINMAX(A, n)
1   i = 1
2   Largest = Smallest = A[i]
3   i = i + 1
4   while i < n
5         if A[i] > Largest
6               Largest = A[i]
7         if A[i] < Smallest
8               Smallest = A[i]
9         i = i + 1
```

Effectiveness

3	8	2	6	9	4	5
A[1]	A[2]	A[3]	A[4]	A[5]	A[6]	A[7]

Swap Numbers

Let **a** and **b** be two numbers. Swap the values of **a** and **b**.

Input: **a**, **b**

Output: **a**, **b** (values swapped)

$\textsc{Swap}(\mathbf{a}, \mathbf{b})$

$1 \quad temp = \mathbf{a}$

$2 \quad \mathbf{a} = \mathbf{b}$

$3 \quad \mathbf{b} = temp$

Effectiveness

5	8

Swap Numbers

Let **a** and **b** be two numbers. Swap the values of **a** and **b**.

Input: **a**, **b**

Output: **a**, **b** (values swapped)

$\textsc{Swap}(\mathbf{a}, \mathbf{b})$

$1 \quad temp = \mathbf{a}$

$2 \quad \mathbf{a} = \mathbf{b}$

$3 \quad \mathbf{b} = temp$

Can you do better?

Effectiveness

5	8

Swap Numbers

Let **a** and **b** be two numbers. Swap the values of **a** and **b**.

Input: **a**, **b**

Output: **a**, **b** (values swapped)

$\textsc{Swap-Efficient}(\mathbf{a}, \mathbf{b})$

$1 \quad \mathbf{a} = \mathbf{a} + \mathbf{b}$

$2 \quad \mathbf{b} = \mathbf{a} - \mathbf{b}$

$3 \quad \mathbf{a} = \mathbf{a} - \mathbf{b}$

Effectiveness

5	8

Analysis

Analysis of an ‘algorithm’ is done with respect to

Space: The total memory required to run the algorithm.

Time: The total time taken by the algorithm to return the output.

Usually, how these two changes with change in input size.

Running time depends on CPU!!

Analyze the asymptotic running time and working space.
Growth of function with respect to input size.
(To what values the running time and the working space will tend to for large input size, while disregarding the system’s configuration.)

Growth of Function

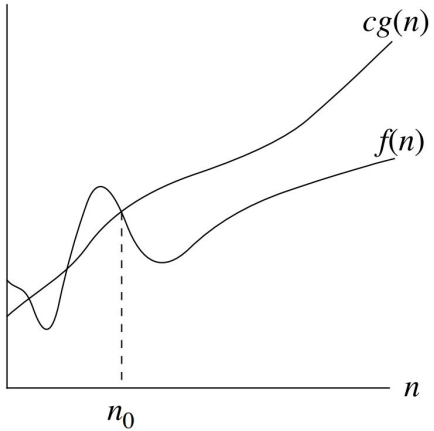
Notations

$$
\begin{array}{ccc}
O & \approx & \leq \\
\Omega & \approx & \geq \\
\Theta & \approx & = \\
o & \approx & < \\
\omega & \approx & >
\end{array}
$$

Growth of Function

O-notation

O(g(n)) = {f(n): there exists positive constants c and n_0 such that $0 \le f(n) \le c \cdot g(n)$ for all $n \ge n_0$}.



g(n) is an asymptotic upper bound for f(n).
So, f(n) = *O*(g(n)).

Growth of Function

O-notation

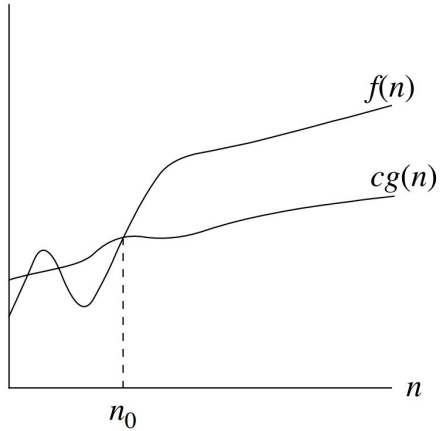
What is the relationship between f(n) and g(n) for the following functions.

- $f(n) = n^2$ and $g(n) = n$
- $f(n) = n^{1/2}$ and $g(n) = n$
- $f(n) = n^{1/5}$ and $g(n) = \log(n)$
- $f(n) = n^{10}$ and $g(n) = 2^n$

Growth of Function

Ω-notation

$\Omega(g(n))$ = {f(n): there exists positive constants c and n_0 such that $0 \leq c \cdot g(n) \leq f(n)$ for all $n \geq n_0$}.



g(n) is an asymptotic lower bound for f(n).
So, f(n) = Ω(g(n)).

Growth of Function

Ω-notation

What is the relationship between f(n) and g(n) for the following functions.

- $f(n) = n^{1/2}$ and $g(n) = n$
- $f(n) = n^{-1}$ and $g(n) = \log(n)$
- $f(n) = n^3$ and $g(n) = 10^n$

Reference

Slides

Introduction to Algorithms by CLRS - Chp-3