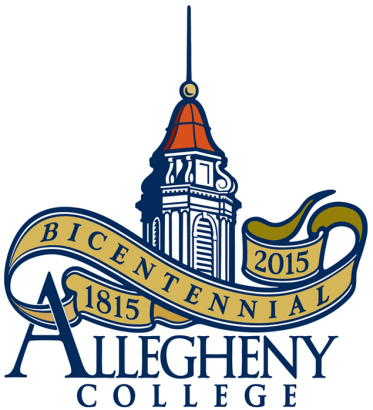
CS202 - Algorithm Analysis

An Introduction

Aravind Mohan

Allegheny College

February 24, 2021



Meeting Time

- Lecture Session:
  - Monday and Wednesday
    9:20 AM - 10:20 AM, (remote)
- Lab Session:
  - Thursday 2:50 PM - 4:40 PM, (remote)
- Practical Session:
  - Friday 9:20 AM - 10:20 AM, (remote)

Professor's Office Hours

- Mondays, Tuesdays, Wednesdays, and Thursdays:
  10:30 AM - 12:00 PM

Email/slack to schedule time outside office hours.

To schedule an office hours time slot, please visit my website [teaching page] and click on the
Schedule Meeting link located on the top right-hand corner to schedule 15 mins slots.
Let us connect with each other and enjoy our time together...

Website Details

- Professor's Website:

  `https://www.cs.allegheny.edu/sites/amohan/`

- Course Website:

  `https://www.cs.allegheny.edu/sites/amohan/course.php?cid=MTM=`

Administrative Stuff!

- No Lab this week.

  First lab next week on Thu, 4th Mar 2021.
- No Class on March 10th, 2021 (college break)
- No Lab on March 25th, 2021 (midterm prep)
- Midterm Exam during lab time, on April 1st, 2021
- Finals at 2:00 PM, on May 17th, 2021 (exam code - A)
- Please verify if you are correctly registered for the course using Self Service.

More Administrative Stuff!

Lab Assignments	**25%**
Skill Tests	**10%**
Midterm Exam	**10%**
Final Exam	**15%**
Course Project	**20%**
Practicals	**10%**
Class Participation	**10%**

Please read the **Syllabus** to get an overview of the course.

Tips for Success

- Attentively listen to classes and try to participate in all class discussions.
- Take detailed notes during every class period.
- Clarify with the Professor, if a lesson is confusing.
- Complete all the reading assignments thoroughly.
- Do the in-class exercises thoroughly.

Be ready to **think**, **process**, and **learn** visually in this course!

Interaction between us...

- Any question is a valid question. There is no question which is good and bad. So, questions are always welcome.
- Interaction is the best way to get rid of long lectures. So, let us try to interact more so that the communication is a two way process and the class is not boring.

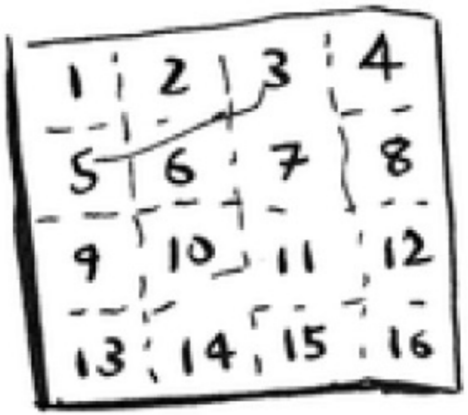
Let us work together to make sure we retain **Algorithmic Knowledge** from this course.

What is an Algorithm?

- An algorithm is a finite set of instructions that if followed, accomplishes a particular task.
- Algorithm is written to make something **fast**, and/or **solve some interesting problems**.

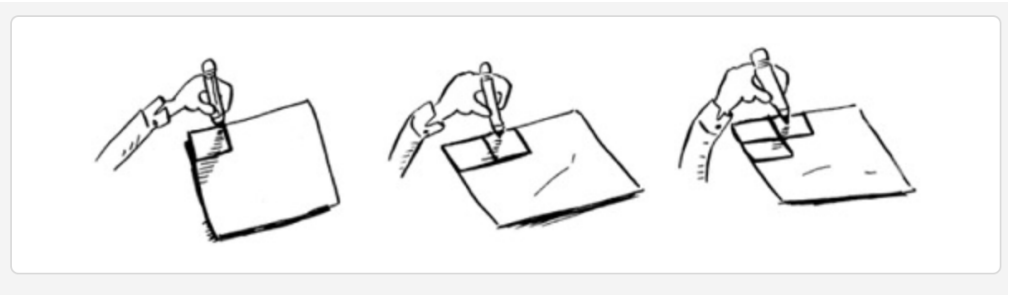
The word Algorithm comes from the name of a Persian author, **Abu Jafar Mohammed ibn Musa al Khowarizmi.**

A Practical Example



- Draw a grid of 16 boxes as shown in the figure above.
- Come up with an Algorithm to solve this problem?

A Practical Example

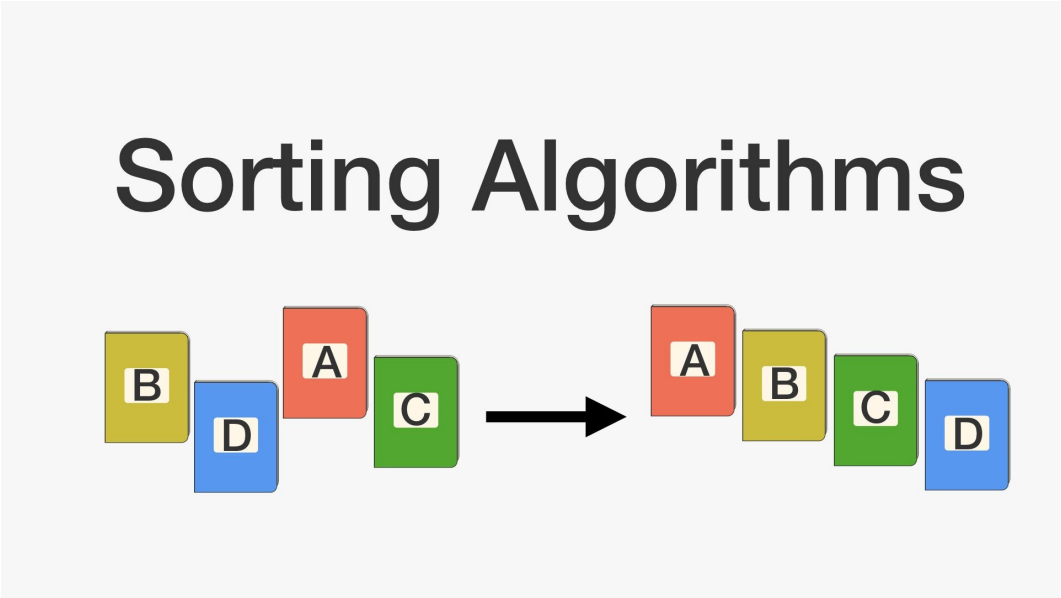


- **Algorithm 1:** One way to do this is to draw 16 boxes, one at a time.
- It takes **16** steps to complete this task.

A Practical Example

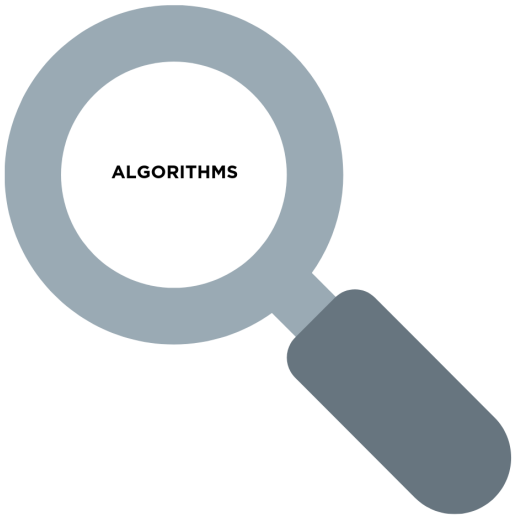
Can we solve this problem faster? **How?**.

What Algorithmic Problems do we learn?



- **Sorting:** For example, sorting music playlist data and rank the songs based on user's interest.
- For example, sorting patient data and rank the patients based on their severity level.
- and more · · ·

What Algorithmic Problems do we learn?



- **Searching:** For example, searching a phone book and finding the phone number associated with a person.
- Searching for a patient's history and finding if the patient had any allergies to medications.
- and more $\cdots$

What Algorithmic Problems do we learn?

- **Graph:** For example, finding the shortest distance between two locations (similar to GPS!)
- Recommend friends based on a person's connections in a friend's network. (similar to Facebook!)
- and more · · ·

What Algorithmic Problems do we learn?

TEXT

- **String Algorithms:** For example, finding longest common subsequence in Strings.
- and more $\cdots$

Learning Goals

By end of this course, you'll in general

- Master a variety of algorithms.
- Be well equipped to learn advanced algorithms in the field of AI, Databases, Cloud Computing, and so on ...
- Be prepared to take on bigger challenges on your senior thesis and at work after graduation.

What do we do in Labs?

- Combination of individual and team-based labs.
- Solve algorithmic problems in a non-programmatic manner.
- Develop algorithms using Programming Languages.
- Compare the performance of algorithms using Charts.

Students may use Python or Java to implement the algorithms.

A More Formal Example

S

156	141	35	94	88	61	111	77
1	2	3	4	5	6	7	8

Algorithm - Find Most Played Song (S, n)
Input - A set of play counts associated with a variety of songs inside a playlist.
Output - The most played song.

```
1: temp ← S[0]
2: res ← 1
3: for i = 1 to n do
4:     if S[i] > temp then
5:         temp ← S[i]
6:         res ← i + 1
7:     end if
8: end for
9: return res;
```

Try out yourself

Algorithm - Find Least Played Song (S, n)

We will try this out during the Practical Session!

Things to do

Think and come up with at least one Algorithm to solve a problem?

- Sign up for course slack channel. (Link accessible in the course webpage!)
- Post your first Slack message. Individually summarize your Algorithmic idea and post a message in the #class-activity channel.
- Read the Syllabus before next class.