Introduction to Algorithms

Science Honors Program (SHP)

Plenary

Christian Lim

Saturday, February 10, 2024

Christian Yongwhan Lim

Education

Stanford University

Part-time Jobs

Full-time Job

Google Research

TWO SIGMA

Workshops

Stanford ENGINEERING | Stanford Computer Forum

Coach/Judge

icpc International Collegiate Programming Contest

icpc.foundation

advancing the art and sport of competitive programming

https://www.yongwhan.io

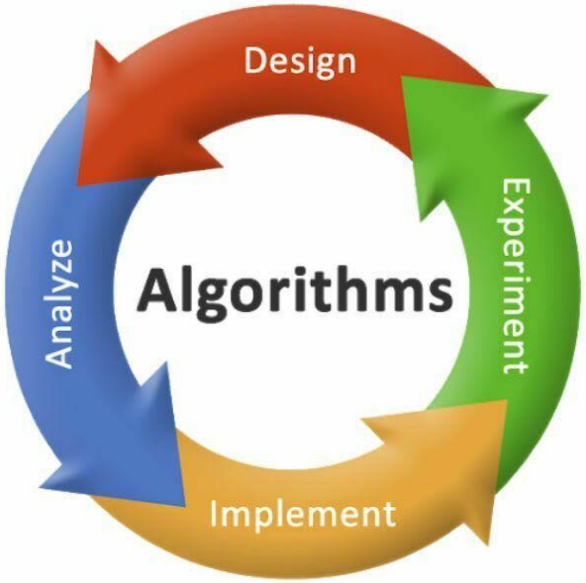
Christian Yongwhan Lim

- Currently:
  - **Adjunct**, Columbia CS;
  - **CEO** (Co-Founder), Stealth Mode Startup;
  - **Co-Founder**, Christian and Grace Consulting;
  - **Head Coach**, Columbia ICPC;
  - **Internship Manager**, ICPC Foundation;
  - **Leadership Team**, ICPC North America (NA);
  - **Trainer**, ICPC NA Programming Camp;
  - **Judge**, ICPC NA Qualifiers and Regionals;

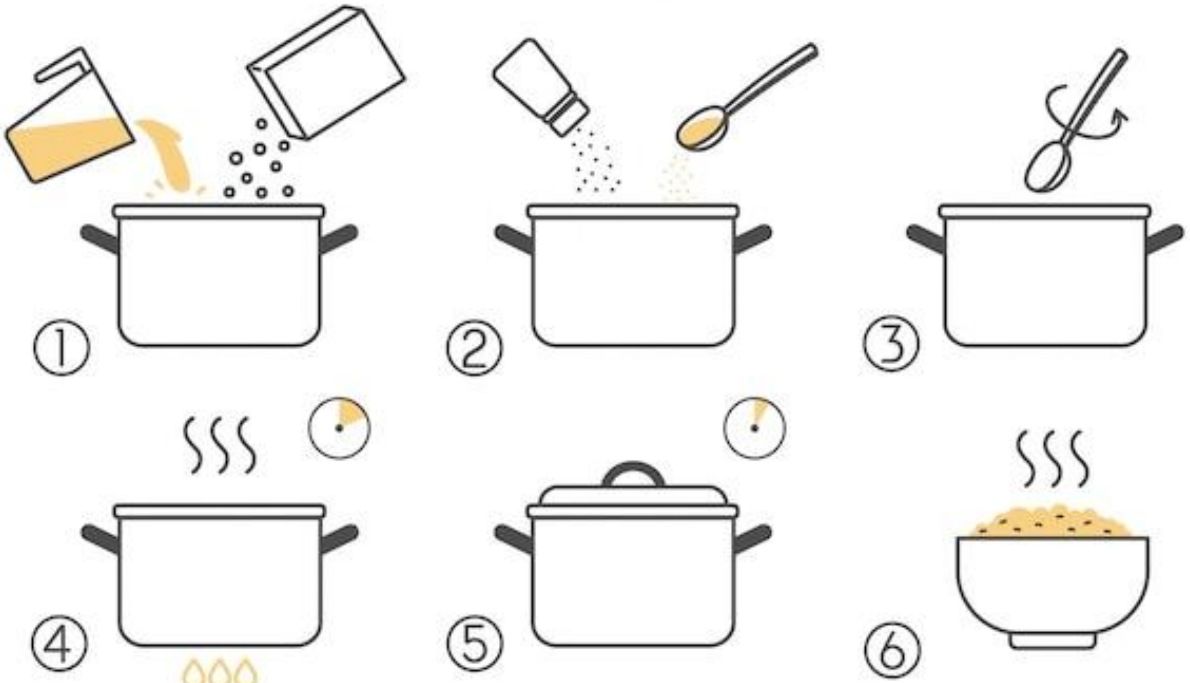
https://www.yongwhan.io

What is Algorithm?

- A set of **step-by-step procedures**, or a set of rules to follow, for completing a specific task or solving a particular problem.



HOW TO COOK PORRIDGE



Why Algorithm?

- **To solve standard problems efficiently!**
  - Typically, efficiency can mean the program runs faster or uses less memory or both!

Why Algorithm?

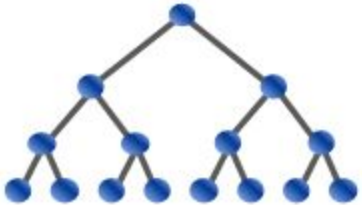
- **To solve standard problems efficiently!**
  - Typically, efficiency can mean the program runs faster or uses less memory or both!
- **To become a better programmer!**
  - This can help you win programming contests!
  - This can help you with an early start on an interview preparation!

Why Algorithm?

- **To solve standard problems efficiently!**
  - Typically, efficiency can mean the program runs faster or uses less memory or both!
- **To become a better programmer!**
  - This can help you win programming contests!
  - This can help you with an early start on an interview preparation!
- **But, most importantly, TO HAVE FUN!**
  - Solving problems can be fun!

USA Computing Olympiad (USACO)

USA Computing Olympiad



If selected, International Olympiad in Informatics (IOI)

Meta Hacker Cup

International Collegiate Programming Contest (ICPC)

Popular Contest Sites

AtCoder

CODEFORCES β
Sponsored by Telegram

topcoder ™

Popular Practice Sites

Popular Tutorial Sites

usaco.guide

cp-algorithms.com

More on Growing Short List of Useful Websites

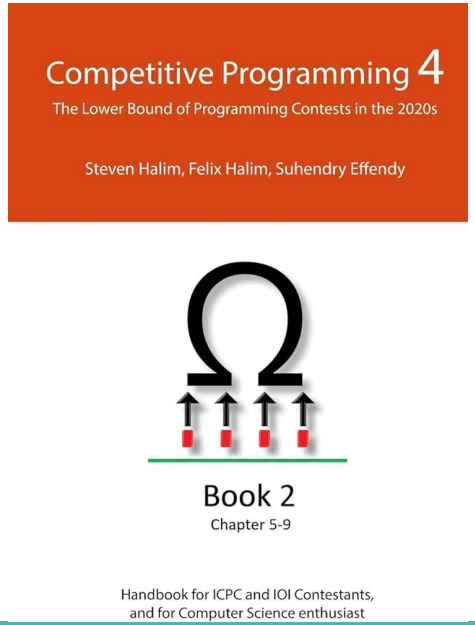
- Please take a look as needed: **Link**

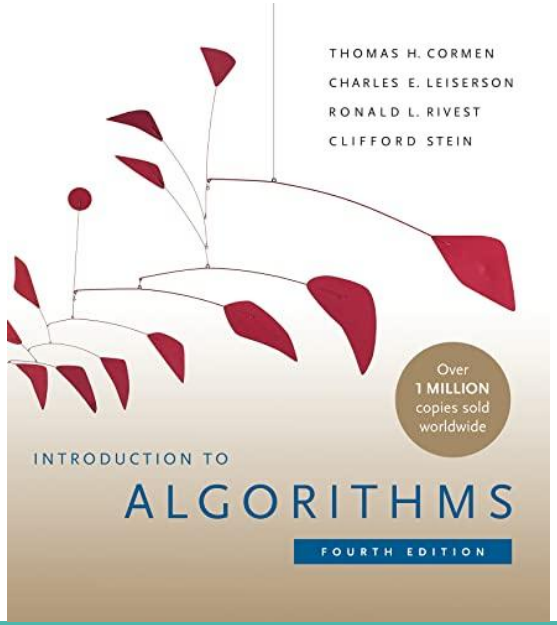
Terse Guides

- Please take a look as needed: **Link**

(Optional) Textbooks

- **Competitive Programming 4**, Halim, et. al.
- **Introduction to Algorithms**, Cormen, et. al.





Programming Language Choice

- You are welcome to pick one of the following languages:
  - **C++**
  - Java
  - Python

- It is the best to pick C++ if you would like to be a serious (competitive) programmer.

Programming Language Background

- Some, if not all, of you may not know how to program yet!
- **This is totally fine/expected!**
- We will build your mastery of C++ ground up.
- If you already know one of the programming languages, that's **awesome**!

Sessions

- We will meet on **Saturdays** from **10am ET** to **12:30pm ET**!

Objectives

- We will go over sampled **topics** from algorithms that are the most essential!

Objectives

- We will go over sampled **topics** from algorithms that are the most essential!
    - **Data Structures** (e.g., C++ Standard Template Library);

Objectives

- We will go over sampled **topics** from algorithms that are the most essential!
  - Data Structures (e.g., C++ Standard Template Library);
  - **Sorting**;

Objectives

- We will go over sampled **topics** from algorithms that are the most essential!
  - Data Structures (e.g., C++ Standard Template Library);
  - Sorting;
  - **Problem Solving Paradigms**;

Objectives

- We will go over sampled **topics** from algorithms that are the most essential!
  - Data Structures (e.g., C++ Standard Template Library);
  - Sorting;
  - Problem Solving Paradigms;
  - **Graph Algorithms**;
    - Shortest Paths;
    - Minimum Spanning Trees;

Objectives

- We will go over sampled **topics** from algorithms that are the most essential!
  - Data Structures (e.g., C++ Standard Template Library);
  - Sorting;
  - Problem Solving Paradigms;
  - Graph Algorithms;
    - Shortest Paths;
    - Minimum Spanning Trees;
  - **String Algorithms**;

Objectives

- We will go over sampled **topics** from algorithms that are the most essential!
  - Data Structures (e.g., C++ Standard Template Library);
  - Sorting;
  - Problem Solving Paradigms;
  - Graph Algorithms;
    - Shortest Paths;
    - Minimum Spanning Trees;
  - String Algorithms;
  - **And, more**!

1:1 Quick Chat

- You may use https://calendly.com/yongwhan/quick-chat-blitz to sign up!

Deliverables

- Nothing!

Deliverables

- Nothing!
- But, there will be in-session, interactive exercises!

ICPC Columbia University Local Contest (CULC)

- **Time:** 2pm ET, One Saturday in April
- **Location:** Somewhere on Columbia University Campus

- Some day in April, we will hold a local contest where all of you are welcome to join, in-person!
- It will be held after lunch on Saturday when the session is held!
- This is an individual, not a team, contest.

Important Note

- Do **NOT** be AFRAID to ask!

- I **LOVE** students asking questions.

- I am here to make you all succeed.

Important Note

- You can think of me as a **coach** rooting for your success in life.

- So, let's pull this through together; I will lead you through this journey!

- **WELCOME** to the world of learning algorithms!

Practice Strategies in CodeForces

- If your goal is to get to a CodeForces rating of **X**, you should practice on problems that are **X + 300** typically, with a spread of 100. So, picking problems within the range of:

  {X + 200, X + 300, X + 400}

  would be sensible!

- So, if you want to target becoming a **red**, which has a lower-bound of 2400, you should aim to solving {2600, 2700, 2800}.
- **(Eventual) Target**: You should focus on solving it for 30 minutes or less!

Practice Strategies in CodeForces

- You should focus on solving each problem for **30 minutes or less**; if you cannot solve any problem with this range, you should consider solving a problem with a lower rating.

- You should aim to solve **10 ~ 15 problems** each day within this range to expect a rank up within a quarter (3 months).

Practice Strategies

- If you cannot solve a problem, here is a sample recipe you can follow:
  - Look at editorial for hints, and try to solve the problem.

Practice Strategies

- If you cannot solve a problem, here is a sample recipe you can follow:
  - Look at editorial for hints, and try to solve the problem.
  - Look at editorial for full solutions, and try to solve the problem.

Practice Strategies

- If you cannot solve a problem, here is a sample recipe you can follow:
  - Look at editorial for hints, and try to solve the problem.
  - Look at editorial for full solutions, and try to solve the problem.
  - Look at accepted solutions, and try to solve the problem.

Practice Strategies

- If you cannot solve a problem, here is a sample recipe you can follow:
  - Look at editorial for hints, and try to solve the problem.
  - Look at editorial for full solutions, and try to solve the problem.
  - Look at accepted solutions, and try to solve the problem.
  - Make sure you look back after two weeks and see if you can solve it.

Live Contest Strategies

- A Terse Guide to Live Contests

C++ Tips and Tricks: best to learn those through practice!

- **C++ Tricks (HosseinYousefi)**
- **C++ tips and tricks (Golovanov399)**
- **Some Tips for Coding in C++ in Competitive Programming (Nea1)**

- Use "**#include <bits/stdc++.h>**" header to include **almost everything**.

ICPC World Finals (@Egypt) in April!

- **Kevin Yang (yangster67)**, **Kaiheng Dai (askd)**, and **Neal Lai (Nea1)** are representing Columbia University in ICPC World Finals at Egypt!

- Due to a travel to Egypt for ICPC World Finals, sessions on April will be substituted by another instructor.

- I will make sure to clarify how it will work closer to the dates. Thanks for your understanding!

Questions and Answers!

- **Ask me anything!**


THANK YOU