

---

---

# **UC San Diego**

# **CP Trainer's Guide to ICPC World Finals**

**Christian Lim**

5pm PT, Friday, May 10, 2024

---

---

# Christian Yongwhan Lim



## Education



## Part-time Jobs



## Full-time Job



## Workshops



## Coach/Judge



<https://www.yongwhan.io>

# Christian Yongwhan Lim



- Christian and Grace Consulting **Owner**;
- ICPC **Internship Manager**;
- ICPC **North America Leadership** Team;
- ICPC **North America Championship** Operations;
- ICPC North America Programming Camp **Trainer**;
- ICPC NAQ and Regionals **Judge**;
- ICPC World Finals CLI Symposium **Co-lead**;
- ICPC North America Curriculum Committee **Co-lead**;
- Columbia ICPC **Head Coach**;
- **Adjunct** (Associate in CS) at Columbia;



<https://www.yongwhan.io>

# Overview

- **Part I: Competitive Programming**
- **Part II: International Collegiate Programming Contest (ICPC)**
- **Part III: World Finals**
- **Part IV: North America Championship (NAC)**

# Part I: Competitive Programming

# Why Competitive Programming?

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

# Why Competitive Programming? (con't)

- **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 Competitive Programming? (con't)

- **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!



# Meta Hacker Cup (YES, recruiting...!)



# International Collegiate Programming Contest (ICPC)



# Popular Contest Sites



# Popular Practice Sites



# Popular Tutorial Sites



[usaco.guide](https://usaco.guide)



[cp-algorithms.com](https://cp-algorithms.com)

# More on Growing Short List of Useful Websites

- Please take a look as needed: [bit.ly/christian-terse-guide](https://bit.ly/christian-terse-guide)
- Alternatively, you can also get to this from [u.icpc.global/training](https://u.icpc.global/training)!

# Programming Zealots @Discord

- Break into **CodeForces** rating of **2200+** as fast as you can!
- Join discord, if you have not already!!!

<https://bit.ly/programming-zealot>



# Programming Zealots @CodeForces

- Also, join CodeForces group, if you have not already!!!

[bit.ly/cf-zealots](https://bit.ly/cf-zealots)





# Success Pathways

- Programming Zealots @ CodeForces
- 800 - 2100 (A - N)
  - **For those who are just starting**
  - To gain some experiences with an explicit goal to enjoy the process of solving new problems;
  - To make it to the ICPC **North America Championship** (NAC)!

# Success Pathways

- [Programming Zealots](#) @ CodeForces
- 800 - 2100 (A - N)
  - **For those who are just starting**
  - To gain some experiences with an explicit goal to enjoy the process of solving new problems;
  - To make it to the ICPC **North America Championship** (NAC)!
- 2200 - 3500 (O - ZB)
  - **For those who are more serious**
  - To make it to the ICPC **World Finals** (and potentially winning a medal)!

# Practice Strategy

- If your goal is to get to a 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 (grandmaster)**, 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 Strategy (con't)

- You should focus on solving each problem for **30 minutes or less**; if you cannot, you should consider solving a problem with a lower rating.
- You should aim to solve **~5 problems** each day within this range to expect a rank up within six months.

## Practice Strategy (con't)

- You should focus on solving each problem for **30 minutes or less**; if you cannot, you should consider solving a problem with a lower rating.
- You should aim to solve **~5 problems** each day within this range to expect a rank up within six months.
- 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 code**, and try to solve the problem.
  - Make sure you **revisit after two weeks** and see if you can solve it.

# Programming Contests

- CodeForces
- AtCoder
- Universal Cup: <https://ucup.ac/register>
- **Quarterly Contests** from ICPC Curriculum Committee, starting **June 2024**

# Training Resources

- **U ICPC:** <https://u.icpc.global/training/>
- **CP Algorithms:** <https://cp-algorithms.com/>
- **USACO Guide:** <https://usaco.guide/>
  
- **Kattis:** <https://open.kattis.com/>
- **Methods to Solve:**  
<https://cpbook.net/methodstosolve?oj=kattis&topic=all&quality=all>
- **CSES:** <https://cses.fi/problemset/>
- **solved.ac:** <https://solved.ac/en>

# Part II: ICPC



# International Collegiate Programming Contest (ICPC)

- If you would like to train as an official ICPC Foundation intern (unpaid), please reach out to me at [christian.lim@icpc.global](mailto:christian.lim@icpc.global).
  - Weekly masterclass on Sundays!
  - Weekly problem set!
  - Weekly 1:1 mentorship!



# International Collegiate Programming Contest (ICPC)

- If you would like to train as an official ICPC Foundation intern (unpaid), please reach out to me at [christian.lim@icpc.global](mailto:christian.lim@icpc.global).
  - Weekly masterclass on Sundays!
  - Weekly problem set!
  - Weekly 1:1 mentorship!
- Or, book a 1:1 time slot using [bit.ly/yongwhan-quickchat](https://bit.ly/yongwhan-quickchat)



# Part III: World Finals

# ICPC World Finals @UCSD

- You can look up the previous results from <https://cphof.org>.
- **UCSD** competed **7** times:
  - **2023** (47th) @ **Luxor**, Egypt: Rank **22** (6/11)
    - Shang Zhou (**ZhouShang2003, 2659**)
    - Stanisław Strzelecki (**stasio6, 2320**): 2017 IOI silver
    - Thomas Li (**bitset, 2138**)
- Prior to this, **UCSD** competed in 2011, 2009, 2005, 2003, 2002, 2001, and 2000.
- You. YES, **YOU(!)** should train hard to get to be the next world finalists!
  - You, and only you, can drive the change here!

# ICPC Training

- So, exactly, **how should you do this?**
  - @Kattis: simulate (**virtually participate**) previous:
    - **World Finals** (WF) problems,
    - **North America Championships** (NAC) problems,
    - **North America Invitation Programming Contests** (NAIPC) problems,
    - Other strong **European** and **Asian** regionals/championships.
  - @ICPC U: look at practice contests from <https://u.icpc.global/training>

# Practice “Strategy”

- The **real keys** to success are:
  - “Upsolving” questions after each session.
  - If solutions are unclear,
    - **STUDY** the algorithms,
    - **IMPLEMENT** them, to make sure you know how to do that,
    - **CHECK** whether you retained them after few weeks,
    - **REPEAT** as many times as needed to learn the algorithms.
  - The **discussions** and **upsolvings** are most probably more important than the simulations! They let you train concepts you DO NOT KNOW!
  - Then, **rinse and repeat** with other problem sets!

# Must topics to master (decider for NAC/WF medals)

- **Discrete Fourier / Number Theoretic Transform** (DFT/NTT)
  - World Finals at Dhaka had a tricky problem using it.
- **Geometry**: New ICPC World Finals judges may put more emphasis on it.
  - sweep-line
  - convex hull (trick)
- **Flow** (Dinic)
- **Mobius Inversion** / Inclusion-Exclusion Principle
- **String**: Aho-Corasick; Lyndon factorization (of course, Z-function, KMP, ...)
- **DP Optimization**: Knuth; Divide and Conquer; Convex Hull Trick;

# Some key learnings from World Finals at Luxor, Egypt

- **Upsolving** is the key for success.
- Identifying the **weakest** topics and iteratively improving are important.
- Holding a **training camp** is crucial to enforce learning fast.
- Solving **Kattis** questions (e.g., WF, NAC, NAIPC), is key.
  - I am in the process of preparing bounty lists to solve.
- Having **multiple coaches** is important.
  - Continue to recruit professors or student coaches.



# Part IV: NAC

# ICPC North America Championship (NAC)

- It started in 2020 for the first time!
  - 2024: To be determined in **May 2024!**
    - Hopeful that UCSD will be TOP 6 for bronze, silver, or gold!
  - 2023: **Rank 5** (solved **9/13**): Bronze
  - 2022: **Rank 31** (solved **4/13**)
  - 2021: **Rank 6** (solved **4/13**): Bronze
  - 2020: **Rank 36** (solved **4/12**)
- Prior to that, there was unofficial contest called **North American Invitational Programming Contest** (NAIPC), which was active from 2014 to 2019!

# 2024 North America Programming Camp (NAPC)

- I am one of the trainers! (<https://www.cecs.ucf.edu/NAC-NAPC/trainers>)
  - **Jingbang Chen** (Waterloo)
  - **Zachary Friggstad** (Alberta)
  - **Andrew He** (MIT)
  - **Ce Jin** (MIT)
  - **Christian Yongwhan Lim** (Columbia)
  - **Quanquan Liu** (Yale)
  - **Etienne Vouga** (UT Austin)
- **You can get more information from** <https://u.icpc.global/training/napc>
  - You should get the training materials from there!

# Any Questions?

- Please find this slide deck from:

<https://github.com/yongwhan/yongwhan.github.io/tree/master/ucsd/2024>

# Contact Information

- Email: [yongwhan@yongwhan.io](mailto:yongwhan@yongwhan.io)
- Personal Website: <https://www.yongwhan.io/>
- LinkedIn Profile: <https://www.linkedin.com/in/yongwhan/>
  - Feel free to send me a connection request!
  - Always happy to make connections with awesome students! :)