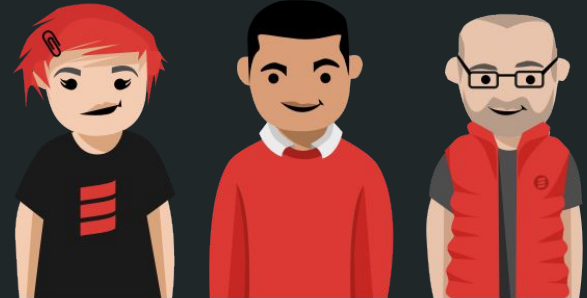


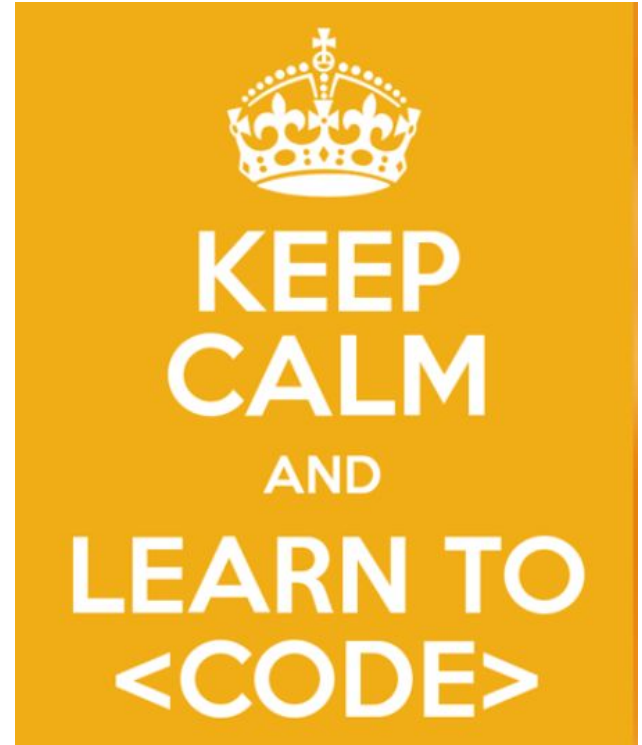
CP & DSA

Mentorship Talks



What all Things you Would Learn?

- A. What is CP & DSA ?
- B. Why to Do CP ?
- C. Online Coding Competitions ?
- D. Languages for CP ?
- E. Where to Start From ? Platforms to Practice ?
- F. What all Contest Should I Give ?
- G. Live Demo
- H. Order in Which Topics to Learn ? Which Topics to Master?



What all Things you Would Learn?

I.) How to Judge Time Complexity from Constraints [TLE Kill]

J.) Template for CP?

K.) Setting up Competitive Programming Environment

L.) CP Books

M.) CP & DSA Websites

N.) YouTube Channels for CP

O.) What is needed to succeed in CP?



What all Things you Would Learn?

P.) What is needed to succeed in CP?

R.) MISTAKES

S.) MISTAKES - II

T.) Golden Nuggets

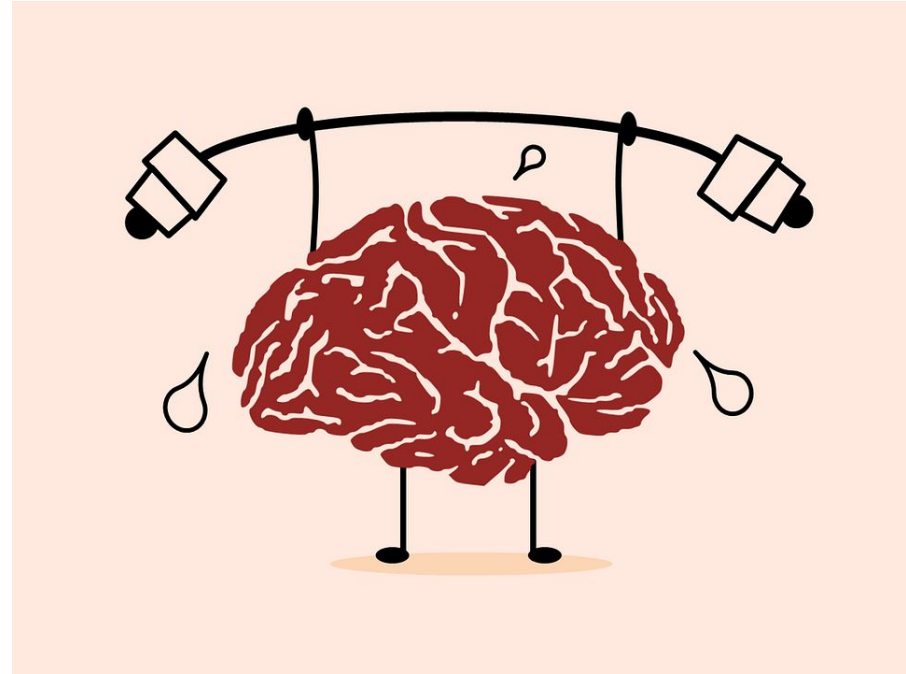
U.) Golden Nuggets - II

W.) Myths About CP

X.) Does Rating Matter?

Y.) THANK YOU FOR ATTENDING !

Z.) ANY QUESTIONS?

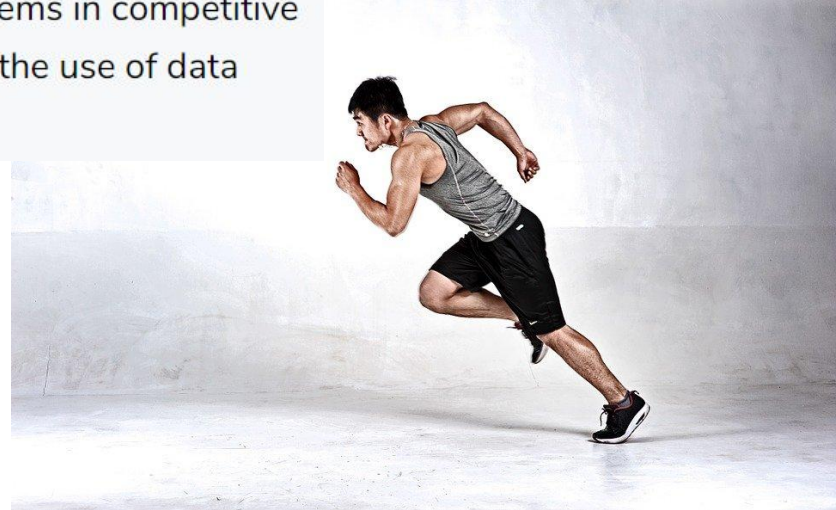


A.) What is CP & DSA ?

Competitive Programming (CP) is a mind sport with one or more logical problems to be solved using programming. Each problem has some specifications which need to be satisfied to solve the problem.

The participants (known as competitive programmers) are required to solve these problems by writing efficient code. Most of the problems in competitive programming are logical or mathematical and may require the use of data structures and algorithms.

It teaches you how to think?



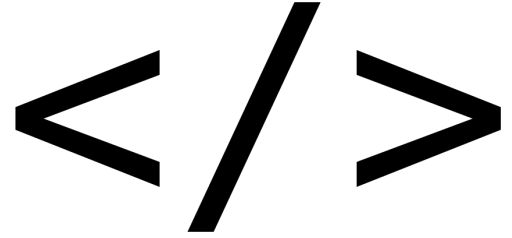
B.) Why to Do CP ?

- ❑ Improve your **Logical and analytical skills**
- ❑ It improves your **Algorithmic** knowledge.
- ❑ It is an excellent addition to your CV.
- ❑ **Improve your Network of Friends** who are also passionate about programming
- ❑ **FANMAG**

- Algorithm Skills
- Basic Math Skills
- Programming Skills
- Typing Speed Skills
- Debugging Skills



B.) Why to Do CP ?



- ❖ Become a **better problem-solver** and a **fast thinker**.
- ❖ Learn **how to write code fast**.
- ❖ Learn how to write **efficient and optimized code**.
- ❖ Learn how to **quickly and effectively debug your code**.
- ❖ Take care of **edge-cases** and become **good at testing**.
- ❖ Learn how to find out the **actual requirements from ambiguous problem statements**.
- ❖ Get a **huge confidence boost and fame**. You might even win some **monetary rewards** or even a **job** through some contests.
- ❖ Get your **resume shortlisted**. If you do really well in Competitive Programming, your resume might help you stand-out as a candidate.

C.) Online Coding Competitions ?

- ❑ **International Collegiate Programming Contest (ICPC)** – for students of universities in groups of 3 persons each
- ❑ **International Olympiad in Informatics (IOI)** - for secondary school students
- ❑ **CodeChef** – 3 contests [Long, Lunchtime, CookOffs] held every month
- ❑ **CodeForces** - 2 Contests/Week
- ❑ **HackerEarth** - 2 [Easy, Circuits]
- ❑ **Atcoder** - [ABC & AGC]





C.) Online Coding Competitions ?

- ❑ **Google Code Jam** is an annual programming competition in which participants are asked to solve complex algorithmic challenges in a limited amount of time.
- ❑ **Google Hash Code** is coding challenge for teams of students and professionals around the world. Top teams will then be invited to a Google office for the Final Round.
- ❑ **Google Kick Start** is a global online coding competition, consisting of three-hour rounds of a variety of algorithmic challenges designed by Google engineers. These competitions are held throughout the year and participants may get a chance to get invited to interview at Google
- ❑ **Facebook Hacker Cup**

D.) Best Languages for Competitive Programming?

A.) **C++** [most recommended and widely used programming languages for Competitive Programming]

- Fastest + Rich Library [**STL**]
- **C++ template classes** that offer various data structures such as lists, trees, stacks, graphs, arrays, queues.
- supports **OOPS** (Object-Oriented Programming) features as well to help you out while solving real-time problems



D.) Best Languages for Competitive Programming?

B.) **JAVA** [Write Once, Run Anywhere]

- ❑ Second Most Preferred Language after C++
- ❑ **Bit Slow**
- ❑ Rich library set - **Collections & Big Integers**
- ❑ **Exceptional Handling and Containers** [OOPS]
- ❑ *All Things which can be Done in C++ can be Done in JAVA* [Long Code]



D.) Best Languages for Competitive Programming?

C.) Python

- High-level programming language is quite **easy to learn** and implement with having a **very simple syntax**
- **Extensive library & function support** which somehow increases the efficiency of programmers in coding competitions
- **No limit on the integer value, functions can return more than one value, code indentation, robust input statements, flexible number of arguments** to a function

Drawback : Some Companies don't Allow Python in Coding Rounds



E.) Where to Start From ? Platforms to Practice ?

Learn Language + Time Complexity

[Programming tutorials, coding problems,
and practice questions](#)

[Learn_DSA](#)

[CP_Tutorial](#)

[GFG](#)



F.) What all Contest Should I Give as Beginner ?

- 1.) [AtCoder](#) Beginner Contest -
Saturday 5:30 PM - 7:10 PM
- 2.) Codechef Starters [Div 3]
- 3.) Codechef Long Challenges
- 4.) HackerEarth Long Challenges
- 5.) CodeForces Contest [Div 3]



G.) Live Demo

- ★ What is Problem Statement, Constraints?
- ★ What are Test Cases?
- ★ What are different Types of Verdict? [AC / WA / RTE / Compilation Error / TLE]
- ★ How are Ranks Decided? [Time Took to Solve & Penalty for WA]

Terrible Chandu



H.) Order in Which Topics to Learn ? Which Topics to Master?

LIST OF TOPICS IN ORDER



I.) How to Judge Time Complexity from Constraints [TLE Kill]

REMEMBER

10^8
operations
per second

By looking at the constraints of a problem, we can often "guess" the solution.

Common time complexities

Let n be the main variable in the problem.

- If $n \leq 12$, the time complexity can be $O(n!)$.
- If $n \leq 25$, the time complexity can be $O(2^n)$.
- If $n \leq 100$, the time complexity can be $O(n^4)$.
- If $n \leq 500$, the time complexity can be $O(n^3)$.
- If $n \leq 10^4$, the time complexity can be $O(n^2)$.
- If $n \leq 10^6$, the time complexity can be $O(n \log n)$.
- If $n \leq 10^8$, the time complexity can be $O(n)$.
- If $n > 10^8$, the time complexity can be $O(\log n)$ or $O(1)$.

J.) Template for CP?

What is Template?

Why to Use it?

[Newbie's C++ competitive programming template. | by Arunkumar Maniam Rajan | Incredible Coder](#)

You can Search “Competitive Programming Template” in Google/YouTube



K.) Setting up Competitive Programming Environment

Sublime

VSCode

You can Modify on Your Own!

[You can Also YouTube “Setup for Competitive Programming”]



L.) CP Books

DATA STRUCTURES AND ALGORITHMS MADE EASY

Data Structures and Algorithmic Puzzles



NARASIMHA KARUMANCHI

M. TECH., IIT BOMBAY, FOUNDER, CAREERMONK.COM

Antti Laaksonen

Guide to Competitive Programming

Learning and Improving Algorithms
Through Contests

Second Edition



 Springer

M.) CP & DSA Websites

<https://cp-algorithms.com/>

<https://www.geeksforgeeks.org/>

Contest Sites -

Codeforces | Topcoder | HackerRank |

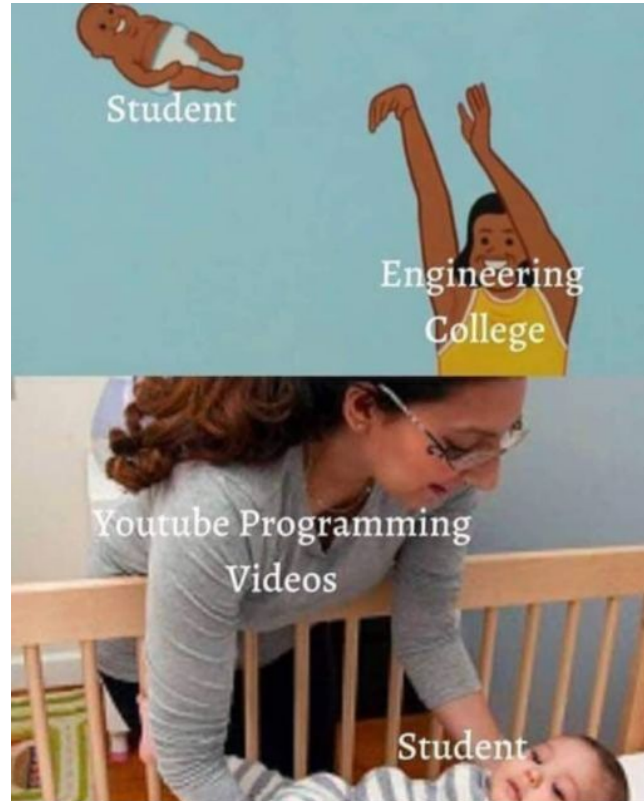
HackerRank | CodeChef | HackerEarth |

CSAcademy | AtCoder | SPOJ |

Leetcode | Project Euler | InterviewBit



N.) YouTube Channels for CP



N.) YouTube Channels for CP



[Prateek Narang \[Coding Blocks\] Tutorials in CP](#) | [Coding Ninjas Tutorials](#)

[PepCoding \[Java\] Sumeet Malik Sir](#) | [Codechef Channel](#)

[Rachit Jain](#) | [Gaurav Sen](#) | [Aditya Verma](#)

[Anuj Bhaiya](#) | [Take U Forward](#) | [Abdul Bari](#)

[C++ | Data Structures and Algorithm](#) [Apna College]

[Code N Code](#) | [Luv](#) | [Hello World](#) | [Love Babber](#) ... List Goes on & On...

[Coding Style of Red Coders](#)

O.) What is needed to succeed in CP?

GRIT

Competitive Programming problems are **difficult** and you may be **stuck on a problem** for a really long time. It requires a lot of grit and perseverance to **sustain** and do well in competitive programming.

PATIENCE

While **Practicing**, you need to be **Patient** enough to solve a lot of problems and spend a lot of time in it **before seeing any significant improvement in contest performance**. Patience is a key skill that you would want to nurture to succeed in competitive programming.



PATIENCE
is the companion of
WISDOM

P.) What is needed to succeed in CP?

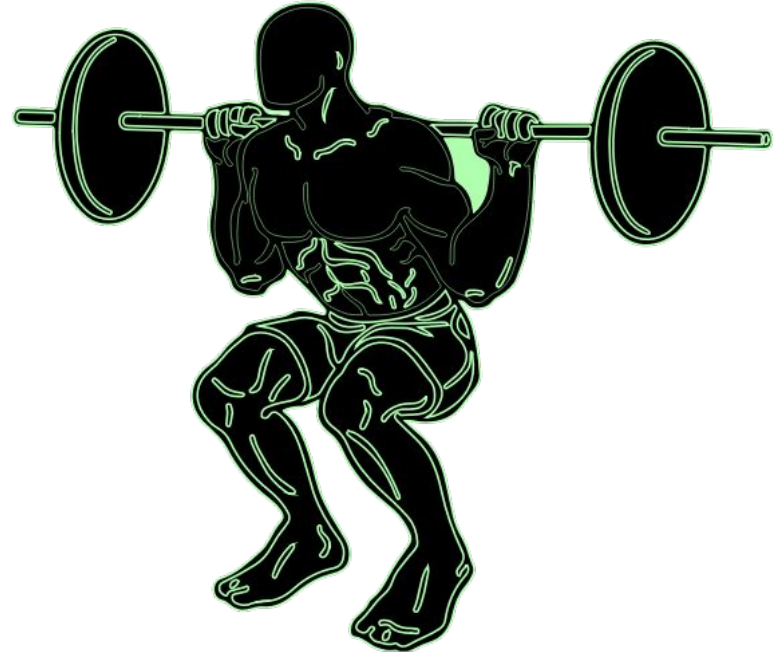
HARD WORK

It requires a lot of hard work for learning, for practicing and even to participate in a lot of contests. **Talent is not enough** to do well in competitive programming.

PRACTICE

PRACTICE

PRACTICE



I'VE FINALLY FOUND
IT... AFTER 15 YEARS

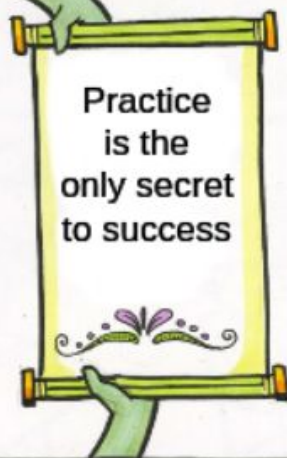


Robotatertotcomics

THE SCROLL OF
TRUTH!



Practice
is the
only secret
to success



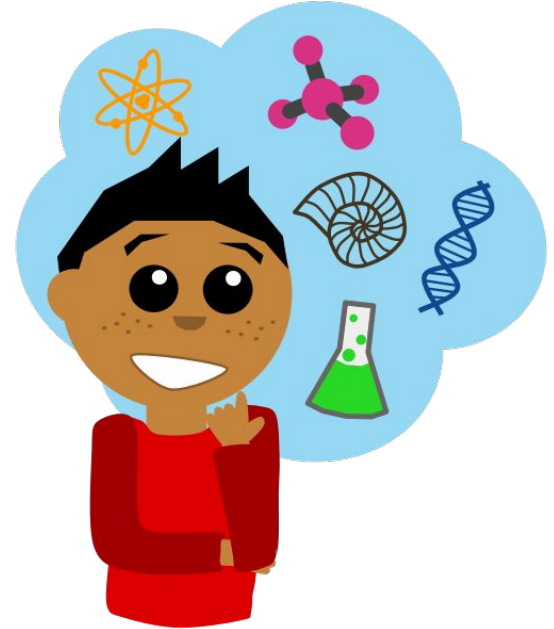
NYEH



NEWBIES, PUPILS AND CYANS

Q.) TIPS

- As a beginner in competitive programming is like giving a **Paper of IIT to 5th or 6th standard student** and then asking him to solve the questions.
- You will be feeling like the same once you know the syntax, loop, function, Data Structures, STL, Algorithm and everything but you are unable to solve those coding problems and that's completely okay. **Understand that every beginner go through this phase.**
- *Higher the submission is, easier the problem is on these coding platforms so you can sort these question by a number of people who solved it.*



R.) MISTAKES

- Programming really needs a lot of **patience**, so don't get discouraged or **DO NOT GIVE UP IF YOU GET WRONG ANSWERS. This is the point where most of the people lose interest in programming.**
- **Keep trying** for a day, week, months if a voice inside your head says that your method is right. If you can't find a solution **google it, ask for help if you don't get the topic, read the various approaches** that coders took to solve the problem but don't quit.



S.) MISTAKES - II

- Also, you can read the solution **only if you are completely sure** that you have made a **lot of effort and now you can't do anything**. But make sure to code again the same problem. **You will start improving day by day only and only if you maintain your consistency and don't stop.**



T.) Golden Nuggets

- ★ **Read the codes of high rated programmers.** Compare your solution with them. You can observe that it is simple and shorter than your solution. Analyse how they have approached and improve your implementation skills.
- ★ **Read the editorials after the contest.** You can learn how to solve the problems that you were not able to solve in the contest and learn alternative ways to solve the problems which you could solve.
- ★ **Always practice the problems that you could solve in the contest.** Suppose if you are able to solve Div 2 250 and 500 in the contest but not Div 2 1000 then practice as many Div 2 1000 problems as as you can.
- ★ **Do not spend too much time** if you are not getting the solution or are stuck somewhere.

U.) Golden Nuggets - II

- ★ After you feel that you have spent enough time, look at the editorials.
Understand the Algorithm and code it Yourself.
- ★ **Do not look at the Actual solution** before you have attempted to write the code on your own.
- ★ Programming is a **very Practical and Hands On** skill. You have to continuously do it to be good at it. It's not enough to solve the problem theoretically, you have to code it and get the solution accepted.
- ★ **Knowing which algorithm/logic to use** and **implementing it** are two different things. It takes both to be good at programming.

W.) Myths About CP

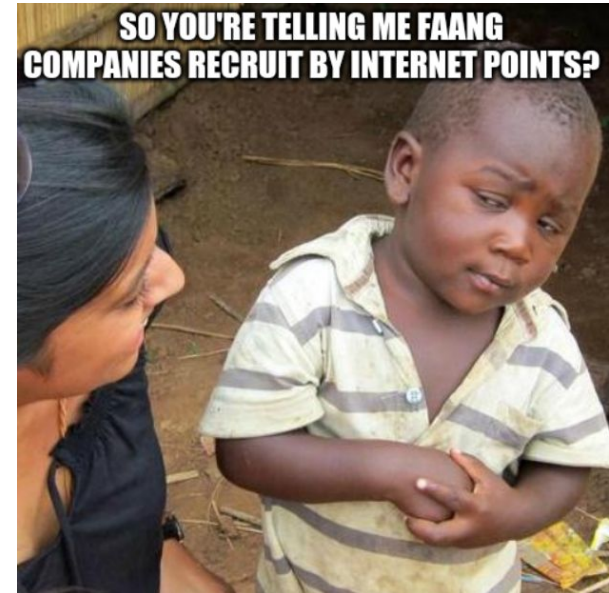
Myths	Reality
It's too late to start competitive programming	There is no fixed age for this best to start earlier in your programming career.
It is an excellent way to get a software programming job	No, it is not true as it is a sport which may benefits. However, it doesn't offer a job guarantee.
You need to solve lots of computing programs before starting competitive programming.	You can learn theory, but you solving computing challenges will not help as every competition is unique with its unique Topcoder challenges.
You need to an expert in algorithm	You need to be able to solve the problems
Competitive programmers are all experienced programmers.	No, it is for everyone even beginner code can participate

X.) Does Rating Matter?

Simple Answer -> **“NO”** [During Interview, No One Asks about Your Rating!]

But it Gives Confidence & Motivation to Keep Improving and Learning!

Your Problem Solving During the Interview is the Thing that Only Counts!



Y.) THANK YOU FOR ATTENDING !

I hope you found the Session
Useful and that I have motivated
you to start coding.



Z.) ANY QUESTIONS?



Programmers while Sleeping



Verdict

Running on test 203



Verdict

Wrong answer on
test 212



MEME TIME



Normal heartbeat



Deceased heartbeat



testing on
test 264

AFTER SOLVING A,B,C



MEME TIME

Coders

Submission
Result

Kids

Accepted ✓

Mens

Wrong Answer ✗

Legends

Time Limit Exceeded ⚠

Super
Legends

Runtime Error !

Ultra
Legends

Compilation Error ⚠



CODEFORCES SYSTEM TESTS EXISTS

**BHAI MAT KAR FAIL YAAR, PURI ROUGH
COPY BHAR GAYI THI SOLVE KERNE MAI**



How C++ looks at Py when it sees $\sim 10^{100}$!



TRAINING

