Note : Take a pen paper and write down important things you find here as it may be a very long thread .

DSA vs Competitive Programming

There is no such difference between them, although there is a difference between standard DSA and competitive programming ..

What do I mean by Standard DSA is the questions that tech companies ask in the OA or interview that you will be able to find on every bhaiya didi sheets

Most of the problems you will do in Competitive programming will never be asked in any OA or interviews..

If no one is gonna ask these problems why to do it and why not do standard DSA .?

To improve logic, simple and that's it .. In tech, the only thing constant is logic.. tech stack will change, Hiring process will change but one thing will always remain the same is logic (your brain)
It not only helps in tech but also makes you think logically in every situation..

Why is logic even important ?

Growing in tech always needs to learn constantly.. If your logic is good, you're gonna adapt things more quickly than others.. This is one of the major reasons why I'm able to score more even studying one night before the semester exam..

What is wrong with Standard DSA and why doesn't it improve logic ?

It improves but a very little coz most of the time you already know which topic is going to be used here and this is almost 90% of the work done .. Another reason is you keep repeating the same type of question again and again, this leads to memorisation and if a question comes out of the standard pattern you won't be able to perform well

Can we improve logic without Competitive Programming ?

A big YES but there is a catch .. you have to try newer questions that don't fit to standard patterns.. Don't get stuck to a roadmap, bhaiya didi sheet(even it's striver A2Z)

Do random questions from leetcode.. Do follow some important standard problems while learning the concept but after some questions from a topic do questions randomly..

NO roadmap, NO pattern, NO topmate sessions, NO bhaiya didi sheet is going to improve your logic.. Only thing that will do is practising new questions

How CP is helpful ?

You give contests where problems are completely new which make you think of all the possible solutions (basically makes your mind to think in all the possible directions) in a time bound manner .. Time is the crucial thing of competitive programming .. a CPer can think a problem in 10 minutes which can take a non CPer 1-2 hours You basically develop your brain to think fast and in right direction

Should I do competitive programming?

If you are desperate to get a job in the next 5-6 months and you haven't studied DSA before, the answer is NO.. CP will take time at least a year from scratch to reach to a certain level so that you can crack a good company .. but solving standard DSA problem may make you familiar with the patterns that is mostly asked in interviews and you may grab an offer with it

So if you haven't studied much DSA yet and need a job in the next 6 months .. this thread is not for you, watch striver's playlist and solve A2Z sheet..

If you have time (at least a year or you have done decent level of DSA and have 5-6 months) yes do CP without a doubt

There is no any Tier bound like I am in Tier 3 should i do CP as in my college company ask very easy questions

Prepare for harder ones, if they ask easy problems you are anyway performing well but if you only prepare for easy problems maybe a chance you won't perform as expected .

Should I do competitive programming or development .?

It’s a tradeoff .. unless you are not a beast you won’t be able to become expert in both CP and dev .. so choose wisely
Be expert in one and above average in the other
Which one should choose should totally depend on what you are enjoying more
If thinking about problems and giving contests give you more thrill go for CP .. else if participating in hackathons or making cool projects give you more thrill go for dev..

How much CP should i do if i want just above average in CP and i am going to be expert in dev?

Atleast knight on leetcode (for clearing interviews) maybe need more for clearing OA

How much CP if I chose CP over dev .?

Atleast expert on codeforces (guardian on leetcode) for clearing most OAs and interviews

Now we are clear if we should do CP or not and if yes how much, let’s proceed how to do it

How to start CP

Step 1 - Learn a programming language (C++ preferably as it is more intuitive and very fast and provides all the required Standard Template Libraries)

Learn it from youtube
Link :- https://youtu.be/e7sAf4SbS_g?si=wkbiVwVDC5wP3oh0

You can choose any video you want but i have learnt few things from Raghav sir and he is quite beginner friendly
Shradha didi from apna college is also good till beginner level
I think Striver is not that much beginner friendly (or maybe my english is not that good)
Choose whichever you feel good

Time required - 10 days

Step 2 - Practise writing codes for some easy problems ..
Link :- https://www.codechef.com/practice/cpp
Time required - 15 days

Step 3 - Learn STL (Standard Template Library)

STL in standard prewritten data structures and functions which you will use mostly everywhere

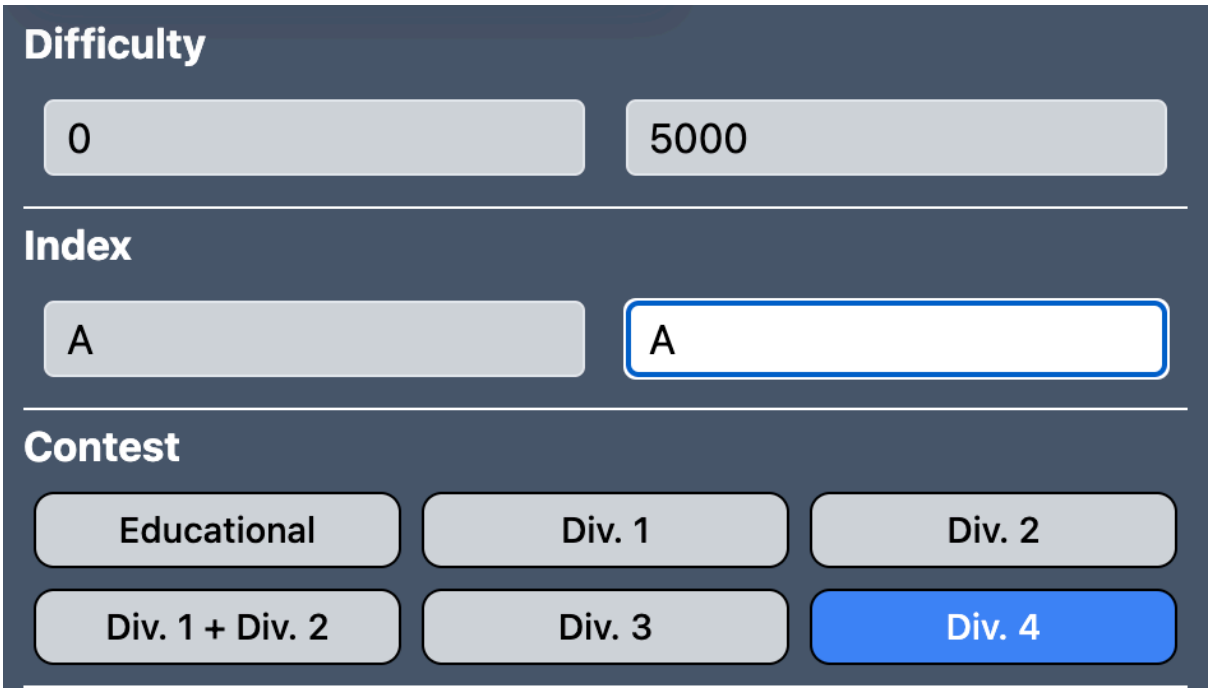
Link :-
https://www.youtube.com/playlist?list=PLauivoElc3gh3RCiQA82MDl-gJfXQQVnn

Time required - 10 days

Step 4 - Practise 50 div3 and div4 As

Link - https://www.acodedaily.com/v2/
Time required - 10 days

Filter problem like below



Now you are ready for giving contest on codeforces

Give every contest you can whether it is div2 or div3 or div4 . don't worry for ratings in the starting phase .. nobody judges you and if any one judge then he will always be inferior than you
I repeat not a single good person judges you so don't worry for ratings at least in initial phases

From rating 0 to 800
What to study
Nothing .. just keep practising questions and giving contests . not a single fancy data structure is required to reach 800 rating .. basic practise on STL and how to think of a problem will take you to 800 easily

Resources to follow
https://www.tle-eliminators.com/cp-sheet (800-1000) rated
Codeforces problemset (800-1000)

You need to solve
Div2 A in the contest
Div3 A,B in the contest
Div 4 A,B faster in the contest

Time required - 2 months

From 800-1200

What to study

- **Basic binary search**
- **Prefix Sum**
- **Sliding window and two pointers**
- **Mathematics**
    - **Exponentiation**
    - **Modular arithmetic**
    - **Number theory**
    - **Primality tests (Sieve)**
    - **Bitwise operations**
- **Time and space complexities and how codeforces judging system works**

Links -

https://www.youtube.com/playlist?list=PL1I6gOernkF5DIhg2mNfIEMtAYuUWagAP

Tip : make your VS code optimised for Competitive Programming and start using templates
https://youtu.be/z4OUph3I6J0?si=_UjSA1NIkp6t0smz

Resources to practise

- **Codeforces Problemset (1000-1400)**
  Do atleast 50 of each range.. If you are not comfortable in 50 go for 100
  Comfortable mean you are able to solve atleast 50-60% problem in a hour without looking at editorial

- **Cses Problemset sorting and searching section**
  Link : https://cses.fi/problemset
  Do 10-15 from start

You need to solve
Div2 A,B in the contest
Div3 A,B,C in the contest
Div 4 A,B,C faster in the contest or Div 4 A,B,C,D in contest
Time required - 3-4 months

From 1200-1400

What to study

- **Binary Search on answers**
- **Dynamic programming**
- **BFS and DFS**
- **Advanced Number theory**
- **Some effective algorithms and techniques (like difference array, line sweep)**
- **Basic Tree operations like how to find subtree sizes in one DFS**

Resources

Striver's DP graph playlist, Priyansh IITGN DP playlist
CSES DP section (7-8 from start)
CSES Sorting and Searching remaining
BFS DFS basic practise from Leetcode
Codeforces Problemset (1400-1600 rating) atleast 50 per rating, if not comfortable then go for 100

You need to solve
Div2 A,B,C in the contest
Div3 A,B,C,D in the contest
Div 4 A,B,C,D faster in the contest or Div 4 A,B,C,D,E

Time required - 4-6 months

Tip :- Start giving Atcoder's Beginner contest (and upsolve till D)

From 1400 - 1600

Congratulations you are better than 90% folks out there and pretty much comfortable to crack any DSA interview
What to study

- **Optimisation of Dynamic Programming**
- **Graph algorithms (Dijkastra, Cycle finding)**
- **Trees (how to find LCA in logn)**
- **Basic Segment tree (or leave it)**
- **Basic String algorithm (Rabin-Karp)**
- **Combinatorics**

Resources

CSES DP
CSES Graph
CSES Trees
Codeforces Problemset (1600-1800) .. same strategy
Atcoder Beginner contest problems D and E

You need to solve
Div2 A,B,C faster in the contest or Div2 A,B,C,D
Div3 A,B,C,D,E in the contest
Div 4 is unrated for you now

Note :- Div2 D may be quite highly rated and 90% of the time you won't be able to solve it. So try to be as quick as possible in div2 A,B,C

Time Required - 6 to 8 months

Some basic Questions

Which question to practise .?

Go to codeforces problemset .. make a filter of your current rating till your rating + 200 .. randomly pick a problem

Sometimes do problems with a tag (atleast for DP)
Use https://codehunt.cc/ for filtering the problems by excluding and including tags . you can also look at what problems other people have done on codeforces on this website

Figure out the topic you are weak at and do 10-15 in your range of that topic.

How much time to think before looking at editorial ?

Think for a problem till you are getting any idea, if you are completely stuck for atleast 15-20 minutes look at the editorial (hints first if given)

Completely stuck doesn't mean thinking for 15-20 minutes .. it means you were getting some ideas for lets say half an hour but nothing worked and now you are blank for 15-20 minutes

Look at editorial line by line and if any idea comes start thinking again for 10-15 minutes

When to learn advanced topics .?

When you encounter a problem which requires any advanced topic, that's the time you need to learn that topic and practise some standard questions on that topic

Where to learn and practise

Youtube (just search the topic name)
Codeforces blogs (for advanced topics which you will rarely don't find on youtube till 1600)
Practise standard on leetcode, cses
https://youkn0wwho.academy/topic-list (another good website, but mostly contain advanced topics, so ignore it till you reach 1500)

I have done basic questions on leetcode, how to start CP .?

Start with leetcode contests, start giving div3s and div4s on codeforces .. you can avoid div2s in the start as you may feel too dumb of yourself even after solving 300-400 leetcode problems .. start giving atcoder beginner contests

You will eventually find out your true rating after 5-6 contest, start grinding now on codeforces

Do I need a course for Competitive programming

Generally NO, but if you are a kind of person who prefers course then you can go for TLE Eliminators run by Priyansh Aggarwal(not sponsored, he is a genuine guy and i have taken the course 6 months ago)
Remember courses are just a helping hand and no course can make you expert on CF, it's just your grind which can ..

How many contests per week .?

Try to do as many as possible(atleast one for sure) .. it depends on person to person, for me after a contest it takes 4-5 hours to recollect myself if the contest goes extremely bad or extremely good (due to cortisol and dopamine respectively)

Atleast don’t miss out codeforces contest..

How to recover from a bad contest .?

Simple answer is just sleep, you can’t do anything .. you will be just overthinking and keep building stress ..
Upsolve the problems next day for sure

How to find the best resources .?

There is nothing as the best resource .. do from wherever you are comfortable
Still listing down some of them

Codeforces blogs (for difficult topics >= 1800 ratings)
CSES Problemset (especially DP, Graph section)
https://acodedaily.com/ (the ultimate ladder)
https://youkn0wwho.academy/topic-list (for searching problems of a particular topic, again for difficult topics)
https://codehunt.cc/ (filtering problems)
https://discord.com/invite/H8TeFjvq6z (A Code daily discord, for connecting with good CPers)

How much time will it take to reach a decent level .?

Assuming you are complete beginner and know nothing about programming

First 2-3 month to learn c++ and stl
3-4 months till 1200
4-6 months from 1200-1400
6-8 months from 1400-1600

So, 1.5 - 2 years to reach expert from very scratch .. if you have some fundamental knowledge of programming it may take lesser .. also depending upon your background and how much IQ you have it may vary

Some general tips

1. If you are from non tier1 stop cribbing, start grinding .. no one is going to give you a job into your plate .. move out of tier3 mindset .. Connect with some tier1 people(on linkedin) and ask them what and how they study

2. Follow right kind of people and just block linkedin influencers who posts things like

Leetcode was hard until i find these patterns
1000 days of code

Don't follow people just because they are in a good org or have high follower count, we all know how hiring was in covid times. Ask LC/CF rating of your influencers

Follow some good accounts on twitter and youtube
Some of them are -
Pranav Mehta
Priyansh Aggarwal
Abhishek Saini
Anu Sharma
Definitely me :-)

3. Don't look for any type of pattern in problems . I can list 100 similar problems where 50 may follow the pattern and 50 won't. Pattern finding is just mugging up

4. Don't focus much on revising the same problem again and again .. after 2-3 times you will start memorising the things and won't use own brain.. Revision leads to memorisation .. do new problems as much as possible .. Revise when interview is just in next month

5. Don’t follow any roadmap, even this thread (just note down important points) .. Simply F*ck around and find out yourself .. roadmap may work for some people and maynot for others and

6. Use leetcode efficiently and don’t care much about your streaks or number of problem counts .. it doesn’t matter, i have leetcode problem count of around 200 and streak of 0 days. Noone gives you a job for problem count or streak count .. you just get a badge of no value
Why not streak - it may seem that you are disciplined but 90% of people just do one problem in the whole day and think they have done coding today

7. Find some friends having same goal as yours and are at almost same level as you (offline is possible , if not then from twitter) .. keep posting updates of your progress on twitter, no one judges you, don’t be shy, you’ll have a catalog and a motivation to study

8. Improve linkedin connections as they are important for requesting referrals if you are not from top college
Just randomly search for top orgs SWE and send connections until your week’s connection request limit exceeds ..

Atleast you should have 20 connection from each org

9. Be clear with your strategy and follow it sometime . don’t just keep changing strategies

10. Do DP, Binary Search alot, you’ll find these in every contest almost

11. Give contest honestly and never cheat. Once you are into cheating there is no way out .. no body cares of your rating .. if your resume says you are expert on codeforces but you can’t find a cycle in a graph .. you’ll be exposed anyway in the interview

12. Upsolve(I’m bad at it too) .. best way to learn from a contest .. upsolving may be quite difficult as you are not sure if the question you weren’t able to do is of your range or not .. but atleast read the problem statement and think for 10-15 minutes

13. Don’t worry for ratings, they will go up and down .

14. Try loving Competitive programming and don’t look at it just as a tool for placement. People generally who love CP(including me) perform better than who do it for placements.

Anyway once you are into Competitive programming and reached a decent level (lets say pupil) you will start loving it and there is no way out form there ..

So the most difficult part is the start .. once you are comfortable with codeforces UI and able to solve atleast div2A, you gonna make CP part of life

15. Atlast, nobody is going to do it for you, it’s your life take its charge into your hands.. Don’t keep crying over anything .. be in a good and motivating circle, keep grinding,almost every CP profile having rating >= 1600 is living a decent life