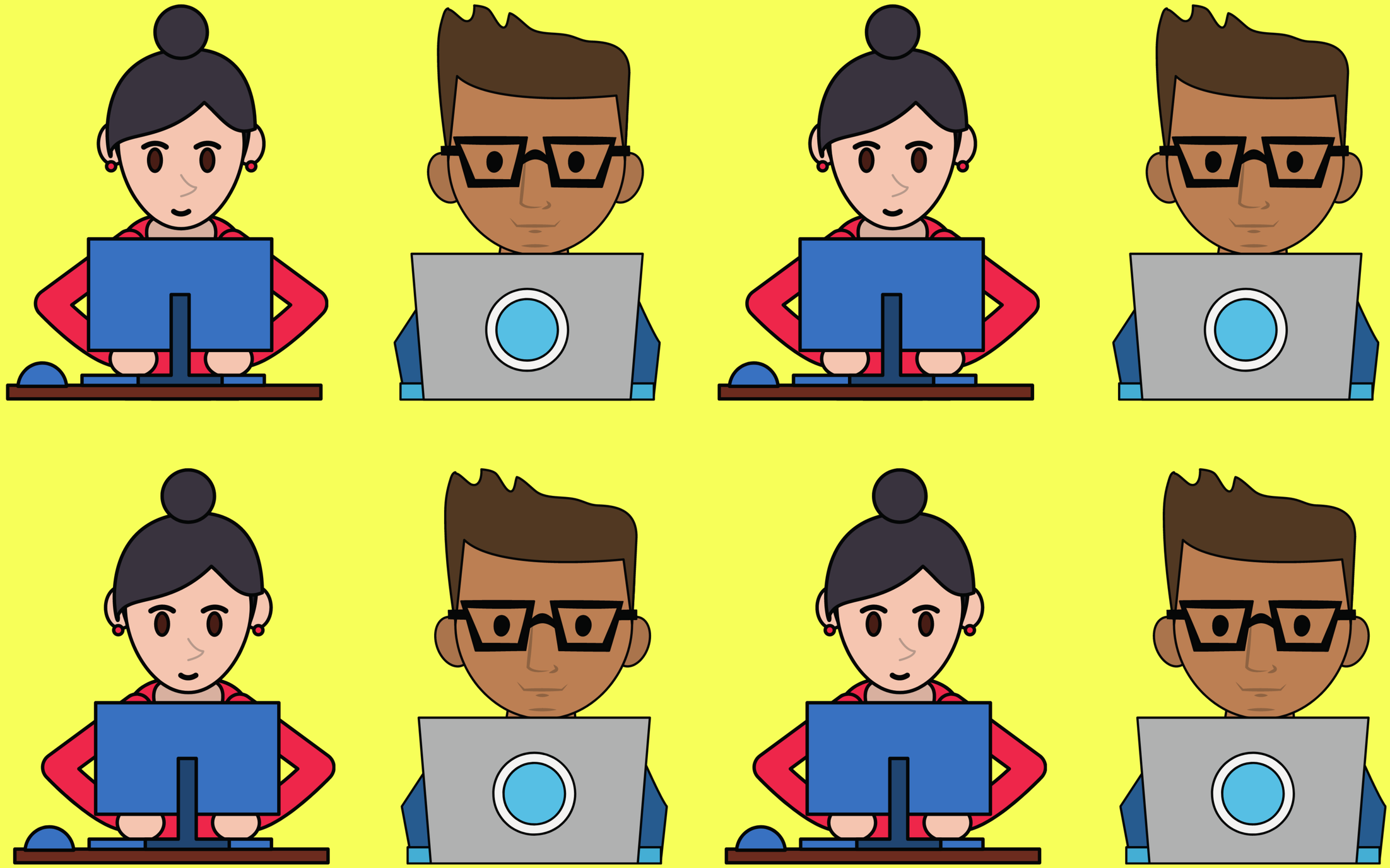
< competitive programming >



WHY COMPETITIVE PROGRAMMING

WHAT IS COMPETITIVE CODING?

- It is basically a mind Sport where you are given some problem and you are required to give an optimized solution under various constraints by using your programming skills.
- This sport basically tests your Logical thinking, Analytical Thinking, Pattern Recognition, Pressure Handling, and most importantly your knowledge of Data Structures and Algorithms.

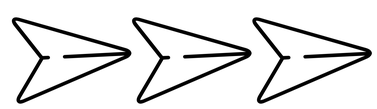
WHY COMPETITIVE CODING?

- You can structure your deliberate practice routine around it
- It will prepare you well for technical interviews
- You'll get used to working on challenging problems
- It's a way to publicly demonstrate your skills
- It's guaranteed brain exercise
- It's a way to focus on the fundamentals
- It's fun

- It's a way to practice fast coding
- You'll really learn your chosen language
- Competitive programming has an active community
- Bite-sized programs are convenient to work on
- Reading (code) is educational

HOW TO GET STARTED?

- Determination / Dedication.
- Choosing your language. Which: The language should be preferably C++ or Java, the last preference should be Python
- Learn the Language
- Understand the Concept of Time and Space Complexity.
- Learn the Fundamentals of Data Structures and Algorithms
- Choose a platform to practice (SPOJ, Codechef, Codeforces, Topcoder, Project Euler....etc)
- Advanced Data Structures and Algorithms
- Practice and Do it Regularly



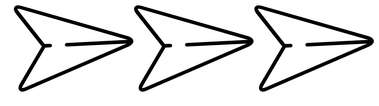
WHAT YOU SHOULD NOT DO

- Don't get Demotivated.
- Not Implementing yourself.
- Do not look into the solution without giving your best.
- Try to re-solve the problem which you were unable to solve in the competetion.
- Only look at solutions after contest ends.
- Talk to people how they are approaching the problem and learn from them.
- Don't stick to easy problems.

NEVER GIVE UP

IMPORTANT TOPICS

- Basic Combinatorial and Number theory
- Sort and Search algorithms
- Hashing
- Number Theory
- Greedy Technique
- Graph Theory
- Disjoint Set Union (Union-find)
- Minimum Spanning Tree
- Segment Tree
- Dynamic programming
- String Algorithms
- Tries, Suffix Tree, Suffix Array.
- Bit-Manipulation - Properties of and, or ,XOR , not gates.
- Geometrical and Network Flow Algorithms



TOP PLATFORMS

topcoder™

Project Euler.net

coderbyte

CODECHEF

An unacademy Educational Initiative

HackerRank

LeetCode

codewars

SPOJ.com

TOP COMPETITIONS

- The ACM International Collegiate Programming Contest (ICPC)
- TopCoder
- Google Code Jam
- The International Conference on Functional Programming
- Microsoft Imagine Cup
- Hewlett Packard (HP) Codewars
- Facebook Hacker Cup
- CodeChef
- Google Summer of Code
- The International Obfuscated C Code Contest
- International Problem Solving Contest
- Google Hash Code
- Google Kick Start