UPANISHAD OF

COMPETITIVE CODING

A BEGINNERS’ MANUAL IN CODING



**PREPARED AND PUBLISHED BY**

ALGORITHMS AND CODING CLUB

IIT DELHI

Contents

Introduction**3**

What is programming/ coding? Why is it so popular?.4

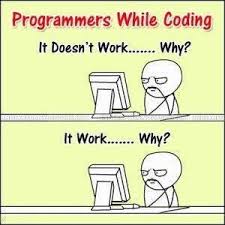
What is competitive coding?.5

Which coding platform suits me? .6

Which language would be the best to program in? …………………………………………7

How to finally get along coding? .........................................................................8

What to do when you get wrong? …………………………………………………………………..9



INTRODUCTION

This work is a compilation of various resources, contributed by the members of Algorithms and Coding Club, IIT Delhi.

The main objective behind this work, is to make your journey of coding comfortable, by providing a plethora of resources to refer to, just a click away, at any point of time.

So dive into this brief manual and end up finding yourself much more informed and confident…



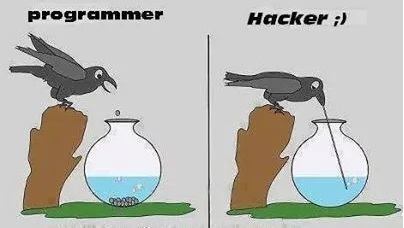
* What is programming/ coding? Why is it so popular?

Programming is the art of solving problems with the aid of a computing machine. Coding is just a small step in the process of programming, wherein one converts an *algorithm* into a computer understandable set of instructions, with the aid of any programming language. An *algorithm* is a set of explicit and unambiguous, finite steps, which when carried out for a given set of initial conditions, produce the corresponding output and terminate in finite time (ref: How to solve it by computer; R. Dromey).

What makes coding popular among people is the fact that it is inherently challenging and exciting. Eventually, it converts into a satisfying experience and gives scope for self- evaluation and improvement. In order to become a good programmer, the first prerequisite is to enjoy the process and to be determined to solve the problem.

For many people, coding becomes important as it an integral part of any job interview.

If you are still sceptical about the benefits you may reap by coding, just refer [here](https://www.codeconquest.com/what-is-coding/benefits/).



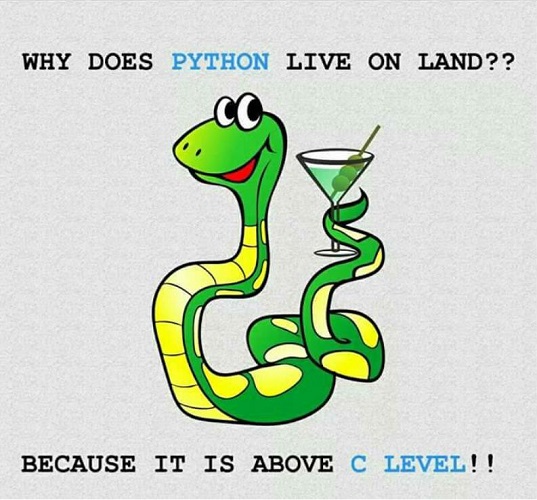
* What is competitive coding?

**Competitive programming** is solving *well-defined problems* by writing *computer programs* under *specified limits*.

For a more involved description, look up the given [post](https://www.quora.com/What-is-competitive-programming-2).

The main motive behind getting involved in competitive coding is to learn coding. This is because coding is all a matter of practice and perseverance, the opportunity for which is provided by competitive coding, in plenty.

How competitive coding can improve your problem solving skills, can be well understood in this [article](https://www.hackerearth.com/practice/notes/kaushik.mv3/learn-to-code-by-competitive-programming/).



* Which coding platform suits me?

Depending upon your experience in competitive coding, you can compete on one of the following platforms:

* **For novices**: Beginning off with [Hackerrank](https://www.hackerrank.com/) or [SPOJ](https://www.spoj.com/) would be a good idea. These platforms provide problems from the basic to advance level with good explanations of solutions. Solving problems on [Geeks for Geeks](https://www.geeksforgeeks.org/) can also help.
* **For scholars**: Continuing off with [Codechef](https://www.codechef.com/) or [Codeforces](http://codeforces.com/) can be a good idea. Practicing on Codechef can help you challenge and improve your understanding. Trying out [Topcoder](https://www.topcoder.com/) also can also be helpful.

Working on [hackerearth or codemonk](https://www.hackerearth.com/blog/community/hackerearth-codemonk-v2-0/), can also be helpful.

Finally, your choice of some coding platform, depends on your taste and comfort. It is always better if one sticks to some platform and practices regularly.

For further reference and guidance on which problems should be solved and general strategies, look up this [article](http://codeforces.com/blog/entry/49157).



* Which language would be the best to program in?

The most important thing is to code correctly and quickly, both in terms of preparing the program and its execution. This requires one to have good command over any language.

In case of beginners, the most popular choices are C++/ Java/ Python. Though all three are good in their respective domains, C++ generally stands out in the competitive coding world due to its less execution time and STL (Standard Template Library).

The pros and cons of each of these languages are clearly articulated in the following posts:

* <https://www.freelancinggig.com/blog/2018/07/20/which-is-better-for-coding-in-algorithms-c-python-or-java/>
* <https://www.geeksforgeeks.org/tips-and-tricks-for-competitive-programmers-set-2-which-language-should-be-used-for-competitive-programming/>



* How to finally get along coding?

“The secret of getting ahead is getting started.”

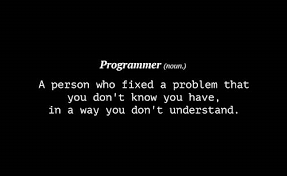
Begin your journey into the world of competitive coding, with the tools of knowledge and practice.

The best way to begin is to choose your language, and some platform and develop your command over the language, the programming paradigms and algorithms dynamically, i.e. through solving topic wise problem sets.

An article to boost your morale to learn more and leave a deep impression on the world of competitive coding is linked [here](https://medium.com/@andreimargeloiu/how-to-prepare-for-competitive-programming-396d557e0c12).

Choose on the mode of learning, with which you are the most comfortable.

* If you prefer reading, then the following tutorials are worth having a look at:
  + <https://www.codechef.com/ioi/basics>
  + <https://www.hackerearth.com/practice/codemonk/> (check codemonk resource)
  + [Book: CLRS](https://eclass.uoa.gr/modules/document/file.php/D21/Introduction.to.Algorithms.CLRS.3rd.Edition.Sep.2009.pdf)
  + [Tutorials with problems](https://www.quora.com/What-basic-data-structures-and-algorithms-should-one-learn-before-starting-competitive-programming/answer/Aman-Goel-9)
* If your preference is video lectures, then the following Youtube channels can be helpful:
  + <https://www.youtube.com/user/tusharroy2525>
  + <https://www.youtube.com/playlist?list=PLMCXHnjXnTnszR6YSo1tQK2BMr15cC9Zh>



* What to do when you get wrong?

When you happen to get your answer wrong, the key is to take a break and introspect. Looking up the test cases again can be helpful.

If you happen to be completely unable to get along with the problem, then is the time to look up the [editorials](https://www.quora.com/Which-programming-contest-site-has-the-best-editorials-for-problems) for the problems and check out any prior submissions.

This will help you to figure out your mistakes and perhaps learn a new programming style.

