cprime

10 STEPS TO BE A BETTER PROGRAMMER

`< / >`

Whether you are just getting into programming or a veteran,
these steps will help you achieve mastery over the craft of programming.

Learn CS Fundamentals — **1**

Learning web development (e.g. JavaScript) is only going to get you so far.

Study fundamentals of CS - Memory allocation, Multi-threading, DataStructures, Algorithms, Complexity Notations etc. It'll alter the way you think about code.

Begin When you Think you Are Done — **2**

So you have working code? Think you are done? That's when you begin with the improvements and refactoring to make the code better and more efficient.

You are only finished when you have nothing else to remove.

Concepts Over Tools — **3**

Understand how to write maintainable and extensible code. Study concepts such as OOP, FP, SOLID and DRY. Delve into Design Patterns. Tools come and go every day but concepts should remain your strength. Breadth of knowledge is far more important than depth.

Think Before You Code — **4**

Train your mind to think like a pro! Pick an interesting problem to solve. The scale and the complexity of the problem at hand don't matter.

Come up with at least couple of approaches on paper. Weigh the pros and cons of each.

Practice. Learn. Perfect. — **5**

There is no substitute for actual keyboard and screen time! Code regularly. Code for yourself, not just for a salary!

Do what makes you happy. Find your interest - Puzzles, Personal project, Open Source project, work-related tasks etc.

Teach. Share. — **6**

You will be a better learner when you teach and share your knowledge with others. To teach others, you will have to first grasp the concepts yourself.

Teach others like you. Or in the local community. Write a blog. Share your learning online. Remember, you don't need to be an expert to get started!

Care About the Quality — **7**

Always strive to improve code. Even if it is fully functional, there will always be room for improvements.

If you are working on a large project, reduce the Technical Debt with every commit. Never write code without backing it up with Unit tests.

Ignore the Hype — **8**

There is a new programming language or a framework every week. Do not try to learn them all. It'll be an overwhelming burden.

Keep up with major technical shifts. They occur only once a decade or so. Cloud computing in the 2000s, Block Chains in 2010s etc.

Acquire Domain Knowledge — **9**

Acquire domain knowledge (The What) in the areas you are working in.

Do this by contributing to Product requirements and participating in Customer engagements.

This knowledge will help immensely in coding the solution (The How).

Read Books! — **10**

Read books, especially those about Good Coding practices, Design Patterns, and Architecture principles. It is recommended that all programmers read "Clean Code" and 'The Pragmatic Programmer".

Bonus Steps

- Be optimistic, don't give up when faced with a seemingly impossible challenge
- Participate in Code Reviews. Try Pair Programming
- Learn a vastly different language than the one you are comfortable with

- Don't reinvent the wheel, look for good libraries that do the job
- Speak at Conferences
- Be comfortable with your IDE
- Don't be afraid to rewrite code

Author: Deepak Karanth