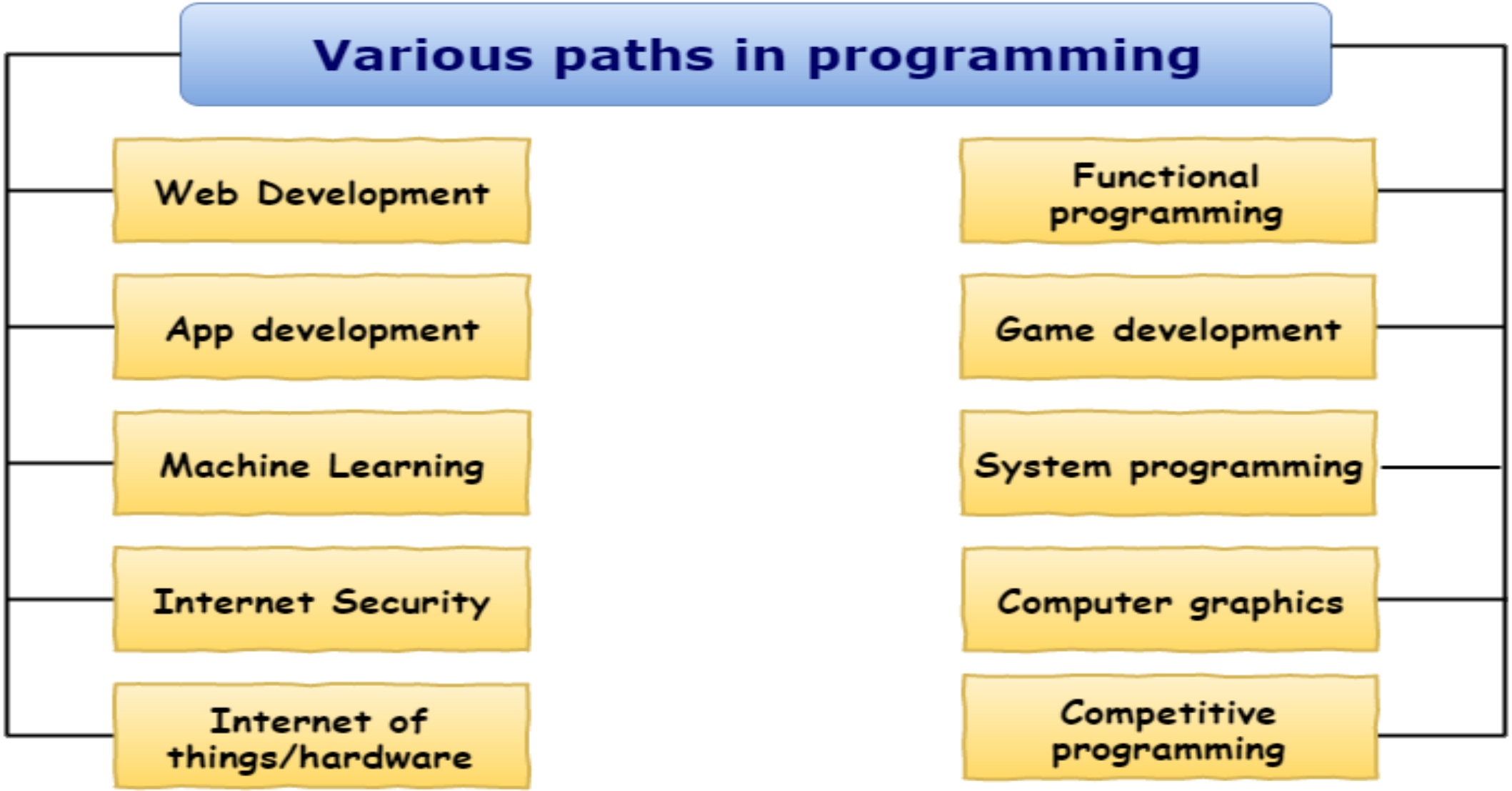
Welcome , freshies

Paths in Programming

By

Aarush Juneja


Various paths in programming
Web Development
App development
Machine Learning
Internet Security
Internet of things/hardware
Functional programming
Game development
System programming
Computer graphics
Competitive programming


WHAT SHOULD I DO NEXT
I HAVE A FEW IDEAS

1. Web development

- Easy to start with
- Lots of things to explore
- 3 paths: Front-end (client-side) ,back-end (server side), full stack (both the ends in depth)
- Basic knowledge like how the web works (networks, ip), HTML, CSS, JavaScript, MySQL, ftp, ssh is expected from every programmer.

1. Web dev. (continued)

Must know basics(Irrespective of front end and back end)

- How the internet works from the client side and the server side
- Basic command line usage
- FTP
- SSH
- HTTP
- What is an API ??
- What are REST/RESTful APIs??

1. Web dev.(continued)

Front End

- HTML5
- CSS3
- JavaScript (e.g. pure js, angular, react, redux, ember etc.)
- Bootstrap (responsiveness)
- jQuery (an extremely popular JS library)

1. Web dev. (continued)

Back end

- PHP
- Python (e.g. Django, Flask)
- JavaScript (e.g. Node.js)
- Ruby(e.g. Ruby on rails)

Some more :

- Lua
- GO

2. Mobile Development

- Android: Java
- Windows (Phones as well as PCs) : C#
- iOS : Objective-C , Swift
- Other very cool and trending options :
    1. Using Python: Kivy
    2. Using JavaScript : Ionic

3. Machine Learning/Data science

The art of giving computers/machines, the ability to think without explicitly programming them to do a specific task.

- Google's Allo, Apple's Siri, Microsoft's Cortana (ML in NLP and human-computer interaction)
- Google search , video recommendation of youtube , Alpha Go, cucumber farm project of a Japanese farmer (Google it!)
- Humanoids

3. M.L. (Cont.)

How to begin with

Programming languages : Python, R etc.

Resources :

1. Python : pythonprogramming.net

2. ML :

a.) Search for “Machine Learning Google developers” on YouTube

b.)Machine Learning course by Stanford University on Coursera

Coming Next is ?? Guess it!!

4. Internet Security

Its all about the ‘cool’ stuff: Cryptography, ethical hacking and all. Too much to discover and loads of fun.

Wanna be a security expert in an industry giant, this field is for you. YES YOU. But nothing in life comes easy. You'll need loads of practice .

You too can become a pro if you PRACTICE basic concepts!!

4. Internet Security (Cont.)

What to do and how to begin with :

E.g. CTFs (Capture The Flag) :

These are security contests that are organized by various colleges , companies etc. to improve/test the knowledge of students in the field of internet security .

The task in all CTFs is to find a FLAG (which is usually string of characters) present in another PC by hacking into it . A lot of hacking softwares and tools, paired with knowledge of cryptography techniques plus too much coolness is what you need.
EXPLORE...

4. Internet Security [Cont.]

Stuff that will get you started

- Before everything , see these 2 links in order :

  1.trailofbits.github.io

  2. github.com/ctfs/resources (Follow this one to keep improving)

- A very famous collection of stuff you will need to be good at CTF

  -> https://github.com/apsdehal/awesome-ctf

5.Hardware

- **Arduino , rasberry pi etc**.

  arduino : embedded **C (**Almost **C** language)

  rasberry pi : **python**
- Visit instructables.com for interesting hardware projects
- For arduino tutorials , visit thenewboston.com or search youtube (there is so much stuff about arduino and rasberry pi over here)
- Separate session planned up for the hardware guys...

6.Functional Programming

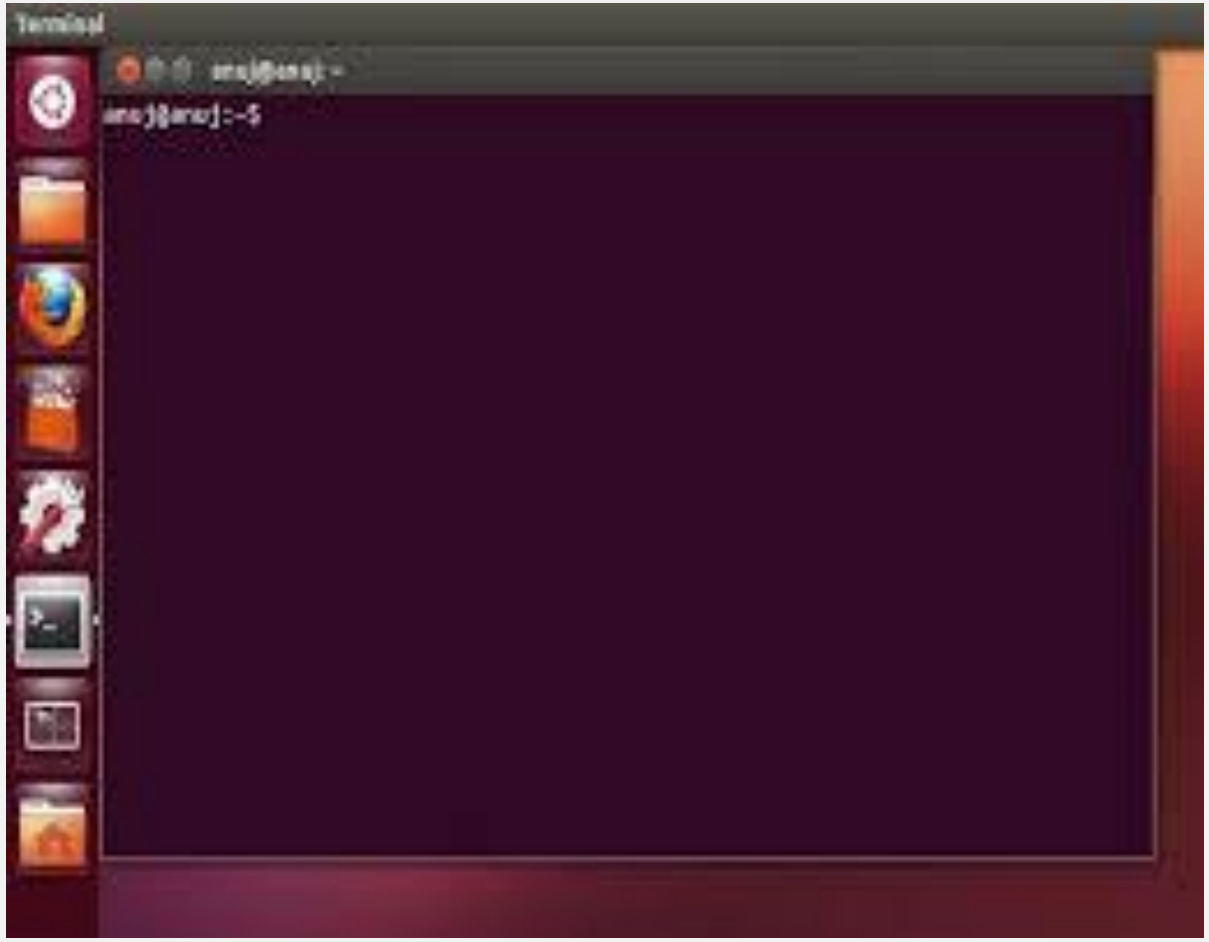
- To be precise , a great technique of programming .
- Languages : Scala , Haskell , F# , Clojure , LISP , OCAML .
- Lets directly talk in terms of numbers...

7. Game development

- Languages you should prefer : C++(openGL) ,C#, Lua .
- For online games : HTML + CSS + Javascript
- Useful software : Unity 3D
- Good practical application of general programming concepts
- Making games will improve your skills a lot in terms of thinking ability and the ability to relate code with what it does.

8. Systems development

- Good knowledge of Bash
- Good knowledge of C
- Good knowledge of shell-scripting



9. Computer Graphics

1. Programming with GLUT and OpenGL (graphic libraries in C)
2. Udacity courses
3. HTML5 Canvas, SVG (for web applications)

   Psst… visit codepen.io once. You'll love it !!
4. Adobe Photoshop/ Illustrator/ After Effects: professional softwares for drawing vector images, editing images etc.

Last but not at all the least . MIND THIS BIG GUN ...

Competitive programming

- How does Google maps find shortest paths very quickly
- Why Google search and not Microsoft's Bing or any other search engine ?? :D :D
- Why Google Chrome is one of the fastest web browser at present?
- Why some shopping apps load search results faster than others??
- How is Facebook able to search a person with mutual friends very quickly as compared to a person with no mutual friends ??

Competitive Programming(Cont.)

Basically, it’s a lot about speed in the cyber world.

The question is, why some softwares/websites/apps run faster than other softwares doing the same thing??(Google search is world's most used engine because of its search speed !! REALIZE THE IMPORTANCE OF SPEED NOW!!)

Under the hood of all softwares, websites , apps, games , etc. is what we call **data structures and algorithms**...

Competitive Programming(Cont.)

Now what are data structures and algorithms ??

Does competive programming teach me those??

Is it okay to just do competitive programming and nothing else at all? (HELL NO!!)

Then why do it ??

List of competitions

1. Microsoft Imagine Cup : Web development , app development , machine learning(NLP , human computer interaction etc.) , functional programming , game development , computer graphics .
2. Microsoft Build the shield : CTF
3. Facebook Capture the flag : CTF
4. InfoSec IITR : CTF hosted by IIT R on their platform(backdoor.sdslabs) ; has participation from around the globe
5. Open Hackerearth anytime and you'll have so many development challenges and online/partially online hackathons
6. HackfestISM : Any sort of hardware or software development

Competitive programming

- Every good programmer recommends C++
- Must begin with C to be thorough with concepts of memory, pointers and other basics
- Beginners : Hackerrank
- 2-3 weeks later : Codeforces , Spoj , Codechef
- After 2 months: Start doing contests on codechef
- Must see hackerearth's famous blog :
  - “Getting started with the sport of programming .”

List of competitive programming competitions

- The most prestigious : ACM ICPC(International Collegiate Programming Contest)
- Google CodeJam
- Facebook HackerCup
- Googla APAC Test
- Codechef Snackdown
- Hackerrank Annual World Codesprint

Open source

- What is open source??
- How does open source benefit you ??
- How to start with it ??
- Would be covered in a separate workshop (if possible)
- You must know git to start with open source properly .
- Biggest oppurtunity in open source: Google Summer of Code (GSoC)