This RoadMap is prepared by:

Saurov Chandra Biswas
Session: 2016-17
Dept of CSE, University of Barishal
Contact: 01881178367
Email : sourav.cse4.bu@gmail.com

Md. Faizul Haque
Session: 2016-17
Dept of CSE, University of Barishal

Email: faizul.cse4.bu@gmail.com

Confidential Rule for the success of a Programmer?

Only One Rule is followed by every Programmer -> Practice, Practice & Practice.

Motivation

Why would I study Computer Science - http://www.shafaetsplanet.com/?p=1639
Why would I learn programming - http://www.shafaetsplanet.com/?p=1437
Programming Contest and Online Judge - http://www.shafaetsplanet.com/?p=1400

BookList

Book - Computer Programming ~ Author - Tamim Shahriar Subeen
Book - Graph Algorithms - Shafaet Ashraf
Book - Data Structures and Algorithms - Coreman
Book - Competitive Programming 3 or 4 - Steven Halim

Newbie(Who has no knowledge about programming)

Start learning C programming (40 days at most)

Reference Book:
https://drive.google.com/file/d/1eQ4NNxzITSWXby2b-H-7ZseewE8SCSDK/view?usp=share_link

Highly recommended to buy a hard copy of this book.
Reference Youtube Channel:
https://www.youtube.com/playlist?list=PLPkEK3TrAJ1M4n273I67kZvz13gsjXPkr

Topics must be covered -

- Data type, Input/ Output
- Operators
- If/else if / else
- Loop - For, While
- Array
- String
- Function
- Structure

Start Solving Problems After learning Loop(20 days)

https://www.beecrowd.com.br/judge/en/problems/index/1
Target Count - 100 problems within 20 days

Beginner(At least have the knowledge of C/C++)(45 days)

Start Learning C++

Topics need to be covered -

- Input/ Output
- String & String functions
- STL(vector, stack, queue, set, map, priority_queue, pair)
  STL resources:
  Website:
  https://www.cplusplus.com/
  PDF:
  https://drive.google.com/file/d/1_oPPOdn4peRpmL4ujykI-APj8HaMmIYQ/view?usp=sharing
  Youtube channel for STL:
  https://www.youtube.com/playlist?list=PLgLCjVh3O6Sgux985GYG22xkFt9z9Sq0_
- Recursion
  Reference video for recursion:
  https://www.youtube.com/watch?v=IxQSirehGP8
- Complexity Analysis
  Ref: http://www.shafaetsplanet.com/?p=1313
  https://drive.google.com/file/d/1dPh1LgDqRQczpTPaneH1I30LsrKKRYwi/view?usp=sharing

Now you are set for solving problems.
Start solving problems from https://codeforces.com/problemset?order=BY_RATING_ASC
Try solving problems sequentially.
Target Problem Count: 200 Problems

Pre-Intermediate(Can solve CodeForces A, B)(90 days)

- Binary Search
  Ref: http://www.shafaetsplanet.com/?p=2279
  https://drive.google.com/file/d/1um5OQ44NdGiQQYaVzRxZ1yQGWj_KfJTM/view?usp=sharing
  Problems on Binary Search:
  https://codeforces.com/problemset?order=BY_RATING_ASC&tags=binary+search
  https://cses.fi/problemset (Sorting Searching Section)

- Two pointer
  https://www.geeksforgeeks.org/two-pointers-technique/
  Problems on Two Pointer:
  https://codeforces.com/problemset?order=BY_RATING_ASC&tags=two+pointers

- STL(unordered_map, multiset,deque)
- Bit Manipulation
  Ref:
  https://www.hackerearth.com/practice/basic-programming/bit-manipulation/basics-of-bit-manipulation/tutorial/
  Problems on BitManipulation:
  https://codeforces.com/problemset?order=BY_RATING_ASC&tags=bitmasks

- Sorting with structure/ pair
  Ref: https://www.tutorialspoint.com/structure-sorting-in-cplusplus
  https://drive.google.com/file/d/1irYBiT9Tf6iu33y5GYlRkUr4PZ9E35eU/view?usp=sharing

- Prefix Sum
  Ref:
  https://drive.google.com/file/d/142B7ZStchmXHLdBL8vUjxG0S6-Sj1_Rg/view?usp=sharing
- Greedy

Ref: https://drive.google.com/file/d/1T6mgESEfcwFLC61QEcruy4irTt4fq_cS/view?usp=sharing

- Basic Geometry
  Problems: https://toph.co/problems/geometry?sort=popularity_desc

- Implementation Problems
  https://codeforces.com/problemset?order=BY_RATING_ASC&tags=implementation
- Target Problem solve Count: 300

How to increase Rating on Codeforces:

First of all, you have to identify your comfort zone. That means you have to identify which problems you can solve more comfortably than others. For example, one might find a 1200 difficulty problem easier to solve but might struggle with 1300/1400 difficulty problem. So he/she should try solving 1400 difficulty problems. Note that, solving more and more 1200-rated problems won't gonna change your skill. You have to choose such problems which will be challenging your brain.
Now how you should approach a problem:
Suppose you have selected a problem that has 200 more difficulty level than your comfort zone, You will try solving this problem for the first 30-45 minutes. If you somehow failed to come out with any logic, then read the first paragraph of the editorial on that problem, don't read the full editorial. Then after taking some insights from the editorial, try solving the problem again. If you again fail to solve that problem, then this time read the full editorial, then try to solve this problem. If you fail again, then this time try watching some code written by famous coders(i.e tourist) then try to implement that code after understanding that code. Believe me, this works like magic.
Participate in at least 1 to 2 virtual contests each day. Try participating from 8:30 to 10:30pm.
A piece of advice, do not engage heavy algo problems until you can solve codeforces A, B and often C.
Watch this one: https://www.youtube.com/watch?v=87oe8kdAjAs

Intermediate(Can solve CodeForces A, B fluently, and C sometimes)(160 days)

- Advanced Greedy
  Ref: https://www.youtube.com/watch?v=IKDtlUMW7F4
- Math, Number theory

Ref:
https://drive.google.com/file/d/1a2KwvopVF955f96u1S3oE0hhD2B1j7v3/view?usp=sharing
https://www.youtube.com/watch?v=ZsZglqx33U8
Gcd, Lcm, Sieve, Prime Factorization, Bigmod, Modular Inverse, NOD, SOD
Sieve: http://www.shafaetsplanet.com/?p=624
Prime Factorization - https://cp-algorithms.com/algebra/factorization.html (trial division)
NOD/SOD - https://cp-algorithms.com/algebra/divisors.html
Problems: https://projecteuler.net/archives
https://lightoj.com/problems/category/modular-arithmetic

- Graph(bfs, dfs, dijkstra, topological sort, disjoint set, Graph Traversal)
  Ref: Book - Graph Algorithms - Shafaet Ashraf
  Book - Data Structures and Algorithms - Coreman
  Book - Competitive Programming 3 or 4 - Steven Halim
  https://drive.google.com/file/d/1Fl76e3d9fmby4LP_Veb0At0M4uxlVTrP/view?usp=sharing
  Dsu - http://www.shafaetsplanet.com/?p=763
  Bfs - http://www.shafaetsplanet.com/?p=604
  Dfs - http://www.shafaetsplanet.com/?p=973
  Graph Traversal Problems: https://toph.co/problems/graph-traversal

- Basic Dynamic Programming(knapsack, coin change, LIS, LCS)
  Ref: https://www.youtube.com/watch?v=cbgdSX2pXcQ (2:17 - Dynamic Programming Basics)
  http://www.shafaetsplanet.com/?p=3638
  Problems on Basic DP: https://atcoder.jp/contests/dp/tasks
  https://codeforces.com/problemset/page/2?tags=dp&order=BY_RATING_ASC

- Sliding Range Minimum Query
  Ref: http://www.shafaetsplanet.com/?p=2316

- Data Structures (Segment tree, pdbs)
  Segment Tree: http://www.shafaetsplanet.com/?p=1557
  https://cp-algorithms.com/data_structures/segment_tree.html

  PBDS: https://www.youtube.com/watch?v=MiBrJTNOEP0
  https://codeforces.com/blog/entry/11080
- Basic Game Theory
  Ref: http://www.shafaetsplanet.com/?p=2325
  http://www.shafaetsplanet.com/?p=2608
  https://www.youtube.com/watch?v=2GoUYpQIAUY

- Interactive Problems
  Ref: https://codeforces.com/blog/entry/45307
  https://www.youtube.com/watch?v=a2QJZT4XDIc
  Problems:
  https://codeforces.com/problemset?order=BY_RATING_ASC&tags=interactive

- Basic Counting and Probability
  Ref:
  https://drive.google.com/drive/folders/1AQrr8LXBO-WHTQ7KYcu08E-1fshg5WN1?usp=sharing (as far as you can)
  Intermediate Higher Math book

- Target Problem Count: 400 problems.

Advanced - 0(Preparation for ICPC/NCPC/IUPC)(300 days)

- Math and Number Theory(Euler Totient, primality tests, etc)
  Euler Totient: https://cp-algorithms.com/algebra/phi-function.html
  Primality tests: https://cp-algorithms.com/algebra/primality_tests.html
  Problems that must be solved: https://lightoj.com/problems/category/number-theory

- Dynamic Programming(Digit dp, Bitmask Dp)
  Digit Dp blog and problems: https://codeforces.com/blog/entry/53960
  Bitmask dp: https://codeforces.com/blog/entry/81516
  Bitmask dp problems: https://lightoj.com/problems/category/bitmask-dp
  Practice problem for DP: Lightoj DP section , Codeforces DP tags(Start from 1700)

- Graph and Tree(MST, Bellman-Ford, Floyd Warshall, Dijkstra Again, Diameter of Tree)
  MST: Book - Graph Algorithms - Shafaet Ashraf
  Book - Data Structures and Algorithms - Coreman
  Book - Competitive Programming 3 or 4 - Steven Halim
  MST Problems: https://toph.co/problems/minimum-spanning-tree

  Bellman-Ford: The same reference can be used as MST.
  Dijkstra & shortest Path Problems: https://toph.co/problems/shortest-path
  Main Objective: **Solve more and more problems from codeforces, lightoj, toph graph section.**

- Tree in-out DP:
  https://returnzerooo.wordpress.com/2018/02/21/%E0%A6%9F%E0%A7%8D%E0%A6%B0%E0%A6%BF-in-out-dp/

- Strings(Hashing, Kmp, Z, Trie(DS))
  Ref: https://www.youtube.com/watch?v=zbV0IRWBNvU
  Hashing: https://cp-algorithms.com/string/string-hashing.html
  Hashing Problems: https://toph.co/problems/tags/hashing?sort=difficulty_asc, https://algo.codemarshal.org/contests/icpc-dhaka-19-preli/problems/A

  Kmp: http://www.shafaetsplanet.com/?p=3209
  https://cp-algorithms.com/string/prefix-function.html
  Z algo: https://cp-algorithms.com/string/z-function.html
  https://medium.com/%E0%A6%AA%E0%A7%8D%E0%A6%B0%E0%A7%8B%E0%A6%97%E0%A7%8D%E0%A6%B0%E0%A6%BE%E0%A6%AE%E0%A6%BF%E0%A6%82-%E0%A6%AA%E0%A6%BE%E0%A6%A4%E0%A6%BE/z-algorithm-string-matching-algorithm%E0%A6%AA%E0%A6%B0%E0%A7%8D%E0%A6%AC-%E0%A7%A6%E0%A7%A7-663527f83131

  Kmp / Z problems - https://codeforces.com/problemset/problem/432/D , https://codeforces.com/problemset/problem/346/B
  https://codeforces.com/contest/291/problem/E(requires KMP optimization)

- Combinatorics
  Ref: https://www.youtube.com/watch?v=fEb_swNH0fY
  Problems: https://toph.co/problems/tags/combinatorics
  https://lightoj.com/problems/category/combinatorics

  Stars and Bars : https://cp-algorithms.com/combinatorics/stars_and_bars.html

- Game Theory
  Ref:
  https://www.youtube.com/watch?v=EienAWnUPow
  http://www.shafaetsplanet.com/?p=2714

- Data Structure(Segment Tree with Lazy Propagation, BIT, Mergesort Tree, LCA, MO)
  Ref: https://www.youtube.com/watch?v=0v--9nEFfAM
  BIT: http://www.shafaetsplanet.com/?p=1961
  Problems on BIT: https://codeforces.com/blog/entry/20569

LCA: http://www.shafaetsplanet.com/?p=1831
https://www.topcoder.com/thrive/articles/Range%20Minimum%20Query%20and%20Lowest%20Common%20Ancestor
Problems on LCA: https://lightoj.com/problems/category/rmq-lca

Segment Tree with Lazy: http://www.shafaetsplanet.com/?p=1591
Merge sort Trees: https://discuss.codechef.com/t/merge-sort-tree-tutorial/14277
Problems on Merge Sort Tree: https://www.codechef.com/tags/problems/merge-sort-tree
https://toph.co/problems/tags/mergesorttree
MO: https://rezwanarefin01.github.io/posts/block-decomposition-01/
MO Problems: https://toph.co/problems/tags/mosalgorithm

- Probability and Expected Value
  Ref: https://www.youtube.com/watch?v=fEb_swNH0fY (follow probability and expected value section)
  http://www.shafaetsplanet.com/?p=3060
  Problems: https://lightoj.com/problems/category/probability
  Problems on Expected values: https://www.codechef.com/tags/problems/expected-value

Advanced - 1(Top 50 in ICPC preliminary)

- Aho corasick
- MO with Update
  Blog: https://rezwanarefin01.github.io/posts/block-decomposition-01/
- Suffix array
  Blog: https://tanvir002700.wordpress.com/2015/01/13/suffix-array/
  Problems: https://toph.co/problems/tags/suffixstructure
- Palindromic tree(eertree)
  Blog: https://rezwanarefin01.github.io/posts/palindromic-tree-01/
  Problems: https://toph.co/problems/palindromic-tree
- Wavelet Tree
- State Space Graph with bfs or dijkstra
  Ref: Book - Competitive Programming 3 or 4
- Manachar
  Blog: https://cp-algorithms.com/string/manacher.html
  Problems: https://codeforces.com/blog/entry/63853
- Mo on Tree
  Blog: https://codeforces.com/blog/entry/43230
  Problems: included on the above blog
- Sack + Small to Large Technique
  Blog: https://codeforces.com/blog/entry/44351
  Problems: Included on the blogs
- Bipartite Graph
- Ternary Search

Advanced - 2(Top 25 in ICPC preliminary)

- HLD
  BLog: https://discuss.codechef.com/t/tutorial-heavy-light-decomposition/69423
  Problems: https://www.codechef.com/tags/problems/heavy-light-decomposition
- Centroid Decomposition
  Blog: https://codeforces.com/blog/entry/81661
  Problems: https://codeforces.com/blog/entry/52492
- Persistent Segment Tree
  Blog: https://rezwanarefin01.github.io/posts/persistent-segment-tree-01/
  https://rezwanarefin01.github.io/posts/persistent-segment-tree-02/
  Problems: https://toph.co/problems/tags/persistentsegmenttree
- Advanced Geometry
- Suffix Automation
  Blog: https://cp-algorithms.com/string/suffix-automaton.html
  Problems: Included on the link mentioned just above.
- Flow Graph
- Fast Fourier Transform
  Blog: https://rezwanarefin01.github.io/posts/fast-fourier-transform/
  Problems: https://www.codechef.com/tags/problems/fast-fourier-transform

Beyond(Top 10 in ICPC preliminary)

Upcoming…

This RoadMap is prepared by:

Saurov Chandra Biswas
Session: 2016-17
Dept of CSE, University of Barishal
Contact: 01881178367
Email : sourav.cse4.bu@gmail.com

Md. Faizul Haque
Session: 2016-17
Dept of CSE, University of Barishal

Email: faizul.cse4.bu@gmail.com