CodeChef Certified Data Structure & Algorithms Programme (CCDSAP)

Next Exam Date
SEP 2019 22

Enroll

Registration Closes on **2nd September 2019**

- Overview
  - Updates
- Levels
- Price & Details
- Schedule
- **Prepare**
  - Foundation
  - Advanced
  - **Expert**
- Examination
- Certified Candidates New
- FAQ
- Contact Us
- Be a Representative
- Dashboard

Learn Data Structures and Algorithms

This section lists out the syllabus, the learning resources and Mock Tests to help you prepare for the Certification test. The resources that we list here are references that we have collected over the internet and some of them from our own website. While we do recommend these resources based on the inputs of our user community, we do not claim that these are the most authoritative Learning Resources about any topic. Please feel free to find out what suits best to you.

We have also prepared a Mock Test for each level. A Mock Test is an open assessment contest that will help you assess yourself for the certification exam after you are ready with the topics. For each level we have different Mock Tests. These contests will run forever. We strongly recommend you to solve these problems in the same duration of time as the duration of the exam before you take the exam.

Candidates can expect problems from the following topics to come in the exam.

Foundation

Syllabus:

The syllabus for each level is mentioned below:

1. Basic Data Structures: Arrays, Strings, Stacks, Queues
2. Asymptotic analysis (Big-O notation)
3. Basic math operations (addition, subtraction, multiplication, division, exponentiation)
4. Sqrt(n) primality testing
5. Euclid's GCD Algorithm
6. Basic Recursion
7. Greedy Algorithms
8. Basic Dynamic Programming
9. Naive string searching
10. O(n logn) Sorting
11. Binary Searching

Learning Resources:

1. **Asymptotic analysis (Big-O notation)**
   a. Basic
      i. youtube.com - Time complexity of a computer program
      ii. youtube.com - Big-O notation in 5 minutes - The basics
      iii. youtube.com - Definition Of Big O Notation - Intro to Theoretical Computer Science
      iv. youtube.com - Algorithms Lecture 1 -- Introduction to asymptotic notations
      v. iarcs.org.in - Measuring the efficiency of algorithms
      vi. interactivepython.org - Particularly for Big-O notation

- Overview
- Updates
- Levels
- Price & Details
- Schedule
- **Prepare**
  - Foundation
  - Advanced
  - **Expert**
- Examination
- Certified Candidates New
- FAQ
- Contact Us
- Be a Representative
- Dashboard

ii. youtube.com - Big O Notation, Gayle Laakman McDowell
iii. web.mit.edu - Big O notation
iv. youtube.com - Time and space complexity analysis of recursive programs - using factorial
v. A very nice tutorial with examples

c. Practice Problems
i. Check some MCQs on space and time complexity here.
ii. You can see some problems with solutions here: Time complexity of an algorithm

2. **Arrays**

a. Resources
i. codechef.com - Data Structure Tutorial: Array
ii. cs.cmu.edu - Arrays
iii. geeksforgeeks.org - Arrays Data Structure

b. Practice Problems
i. codechef.com - LECANDY, editorial
ii. codechef.com - CNOTE, editorial ;
iii. codechef.com - SALARY, editorial
iv. codechef.com - CHN15A, editorial
v. codechef.com - RAINBOWA, editorial
vi. codechef.com - FRGTNLNG, editorial
vii. codechef.com - COPS, editorial

3. **Strings**

a. Resources
i. tutorialspoint.com - C++ strings
ii. guru99.com - Java strings
iii. docs.python.org - Python strings
iv. tutorialspoint.com - Python strings
v. geeksforgeeks.org - Many string questions

b. Practice Problems
i. codechef.com - CSUB, editorial
ii. codechef.com - LAPIN, editorial

4. **Stack and Queue**

a. Resources
i. geeksforgeeks.org - Stack Data Structure
ii. geeksforgeeks.org - Introduction and Array Implementation
iii. tutorialspoint.com - Data Structures Algorithms
iv. cs.cmu.edu - Stacks
v. cs.cmu.edu - Stacks and Queues
vi. cs.cmu.edu - Stacks and Queues

b. Practice Problems
i. spoj.com - JNEXT
ii. spoj.com - STPAR
iii. spoj.com - ONP
iv. codechef.com - COMPILER
v. spoj.com - MMASS
vi. spoj.com - HISTOGRA
vii. codeforces.com - D. Maximum Xor Secondary
viii. spoj.com - ANARC09A
ix. codeforces.com - C. Minimal string
x. codeforces.com - B. Alternating Current

5. **Basic math operations (addition, subtraction, multiplication, division, exponentiation)**
   a. codechef.com - A tutorial on Fast Modulo Multiplication
6. **Euclid's GCD Algorithm**
   a. Resources
      i. youtube.com - Mycodeschool video
      ii. khanacademy.org - The Euclidean Algorithm
      iii. geeksforgeeks.org - Example program to find gcd in c++:
7. **Prime Numbers, divisibility of numbers**
   a. Resources:
      i. Only O(sqrt(n)) algorithm for finding whether a number is a prime, factorization of a number.
      ii. Finding prime factors by taking the square root
   b. Practice Problems:
      i. community.topcoder.com - DivisorInc
      ii. community.topcoder.com - Prime Polynom
      iii. community.topcoder.com - Prime Anagrams
      iv. community.topcoder.com - Refactoring
8. **Basic Recursion**
   a. Resources
      i. topcoder.com - An Introduction to Recursion, Part 1
      ii. topcoder.com - An Introduction to Recursion: Part 2
      iii. geeksforgeeks.org - Recursion ;(along with questions)
      iv. web.mit.edu - Recursion
      v. csee.umbc.edu - Recursion ;(Examples with exercises)
      vi. loveforprogramming.quora.com - Backtracking, Memoization & Dynamic Programming
      vii. byte-by-byte - Recursion for Coding Interviews
   b. Practice Problems
      i. codechef.com - NOKIA, editorial
      ii. codechef.com - TRISQ, editorial
      iii. codechef.com - LFSTACK, editorial
      iv. codechef.com - FICE, editorial
9. **Greedy Algorithms**
   a. Resources
      i. iarcs.org.in - Greedy Algorithms
      ii. iarcs.org.in - Greedy Algorithms
      iii. topcoder.com - Greedy Algorithms
      iv. Greedy Algorithms
   b. Practice Problems
      i. codechef.com - TACHSTCK, editorial
      ii. codechef.com - CIELRCPT, editorial
      iii. codechef.com - MAXDIFF, editorial
      iv. codechef.com - CHEFST, editorial
      v. codechef.com - CAKEDOOM, editorial
      vi. codechef.com - CLETAB, editorial
      vii. codechef.com - TADELIVE, editorial
      viii. codechef.com - MANYCHEF, editorial
      ix. codechef.com - MMPROD, editorial
      x. codechef.com - CHEFTMA, editorial
      xi. codechef.com - STICKS, editorial
      xii. spoj.com - RAISED

- Overview
- Updates
- Levels
- Price & Details
- Schedule
- **Prepare**
  - Foundation
  - Advanced
  - **Expert**
- Examination
- Certified Candidates New
- FAQ
- Contact Us
- Be a Representative
- Dashboard

- Overview
- Updates
- Levels
- Price & Details
- Schedule
- **Prepare**
  - Foundation
  - Advanced
  - **Expert**
- Examination
- Certified Candidates New
- FAQ
- Contact Us
- Be a Representative
- Dashboard

xv. codechef.com - FGFS
xvi. codechef.com - KNPSK
xvii. codechef.com - LEMUSIC
xviii. spoj.com - ARRANGE
xix. spoj.com - FASHION

10. **Dynamic programming (Basic DP)**
    a. Resources
        i. medium.freecodecamp.org - Demystifying Dynamic Programming
        ii. iarcs.org.in - Dynamic Programming - Tiling
        iii. topcoder.com - Dynamic Programming – From Novice to Advanced
        iv. illinois.edu - Dynamic Programming ;(Exercises are recommended)
        v. codechef.com - Dynamic Programming
        vi. geeksforgeeks.org - Dynamic Programming ;(Contains a lot of practice sessions)
        vii. MIT OCW (Contains some Advanced topics as well)
            i. Dynamic Programming I
            ii. Dynamic Programming II
            iii. Dynamic Programming III
            iv. Dynamic Programming IV
    b. Practice Problems
        i. codechef.com - ALTARAY, editorial
        ii. codechef.com - DELISH, editorial
        iii. codechef.com - DBOY, editorial
        iv. codechef.com - XORSUB, editorial
        v. codechef.com - GRID, editorial
        vi. codechef.com - TADELIVE, editorial
        vii. codechef.com - FROGV, editorial
        viii. codechef.com - MATRIX2, editorial
        ix. codechef.com - AMSGAME2, editorial
        x. spoj.com - MDOLLS
        xi. spoj.com - MSTICK
        xii. spoj.com - MCARDS
        xiii. spoj.com - MIXTURES
        xiv. spoj.com - SAMER08D
        xv. spoj.com - AIBOHP

11. **Naive string searching**
    a. Resources
        i. geeksforgeeks.org - Naive Pattern Searching

12. **Sorting**
    a. khanacademy.org
    b. visualgo.net
    c. iarcs.org.in
    d. Merge sort
        i. youtube.com - Merge sort algorithm
        ii. Practice Problems
        codechef.com -MRGSRT
    e. Quick sort
        i. youtube.com - Quicksort algorithm
        ii. Practice Problems
        codechef.com -TSORT

i. codechef.com - TACHSTCK, editorial
ii. codechef.com - STICKS, editorial

13. **Binary Search**
    a. Resources
        i. topcoder.com (Try solving problems of Simple and Moderate level as mentioned in the end of the link)
        ii. codechef.com
        iii. usfca.edu
        iv. khanacademy.org
    b. Detailed Theoretical analysis
        i. cmu.edu (A theoretical analysis)
    c. Problems
        i. geeksforgeeks.org - Binary Search (Contains some solved problems)
        ii. codechef.com - STRSUB, editorial
        iii. codechef.com - ASHIGIFT, editorial
        iv. codechef.com - STACKS, editorial
        v. codechef.com - DIVSET, editorial
        vi. codechef.com - LOWSUM, editorial
        vii. codechef.com - SNTEMPLE, editorial
        viii. codechef.com - SNAKEEAT, editorial
        ix. codechef.com - SCHEDULE, editorial
        x. codechef.com - RIGHTTRI, editorial
        xi. codechef.com - FORESTGA, editorial
        xii. codechef.com - CHEFHCK2,editorial
        xiii. spoj.com - ABCDEF
        xiv. spoj.com - NOTATRI
        xv. spoj.com - SCALE
        xvi. spoj.com - SUMFOUR
        xvii. spoj.com - SUBSUMS
        xviii. spoj.com - ANARC05B
        xix. spoj.com - RENT
        xx. spoj.com - PIE
        xxi. spoj.com - MKUHAR
        xxii. spoj.com - SVADA
        xxiii. spoj.com - SUBS

Mock Test:

1. Test 1 - codechef.com/FLMOCK01
2. Test 2 - codechef.com/FLMOCK02
3. Test 3 - codechef.com/FLMOCK03
4. Test 4 - codechef.com/FLMOCK04

Advanced

This level is intended to test that the candidate has a very good grasp of algorithms and data structures, and can solve most problems that arise in practice. Candidates can expect problems from the following topics to come in the exam.

Syllabus:

Everything in the Foundation Level, along with:

Overview
Updates
Levels
Price & Details
Schedule
Prepare
Foundation
Advanced
Expert
Examination
Certified Candidates New
FAQ
Contact Us
Be a Representative
Dashboard

3. Segment Trees
4. Binary Index Tree (Fenwick tree)
5. Trees (traversals, tree dynamic programming)
6. Finding Lowest Common Ancestors (O(log N) solution where N is number of nodes).
7. Graph Algorithms:
    a. Finding connected components and transitive closures.
    b. Shortest-path algorithms (Dijkstra, Bellman-Ford, Floyd-Warshall)
    c. Minimum spanning tree (Prim and Kruskal algorithms)
    d. Biconnectivity in undirected graphs (bridges, articulation points)
    e. Strongly connected components in directed graphs
    f. Topological Sorting
    g. Euler path, tour/cycle.
8. Modular arithmetic including division, inverse
9. Amortized Analysis
10. Divide and Conquer
11. Advanced Dynamic Programming problems (excluding the dp optimizations which are added in expert level)
12. Sieve of Eratosthenes

- Overview
- Updates
- Levels
- Price & Details
- Schedule
- **Prepare**
    - Foundation
    - Advanced
    - **Expert**
- Examination
- Certified Candidates New
- FAQ
- Contact Us
- Be a Representative
- Dashboard

Learning Resources:

1. **Heaps (priority queue)**
    a. Resources
        i. cs.cmu.edu
        ii. eecs.wsu.edu
        iii. geeksforgeeks.org
        iv. visualgo.net
        v. iarcs.org.in
    b. Practice Problems
        i. codechef.com - IPCTRAIN, editorial
        ii. codechef.com - ANUMLA, editorial
        iii. codechef.com - KSUBSUM, editorial
        iv. codechef.com - RRATING, editorial
        v. codechef.com - TSECJ05, editorial
        vi. spoj.com - WEIRDFN
        vii. codechef.com - CAPIMOVE, editorial
        viii. spoj.com - RMID2
        ix. spoj.com - LAZYPROG
        x. spoj.com - EXPEDI
        xi. acm.timus.ru
        xii. baylor.edu - Maze Checking and Visualization
        xiii. codechef.com - MOSTDIST, editorial
2. **Disjoint Set Union**
    a. Resources
        i. topcoder.com
        ii. harvard.edu
        iii. ucdavis.edu
        iv. visualgo.net
    b. Practice Problems
        i. codechef.com - GALACTIK, editorial
        ii. codechef.com - DISHOWN, editorial
        iii. codechef.com - JABO, editorial

vi. B. Mike and Feet
vii. D. Quantity of Strings
viii. codechef.com - SETELE, editorial
ix. codechef.com - MAZE, editorial
x. codechef.com - MAGICSTR, editorial
xi. codechef.com - MTRWY, editorial
xii. codechef.com - BIGOF01, editorial
xiii. codechef.com - FIRESC, editorial

3. **Segment Trees**
   a. Resources
      i. wcipeg.com
      ii. topcoder.com
      iii. kartikkukreja.wordpress.com
      iv. visualgo.net
      v. iarcs.org.in
   b. Practice Problems
      i. spoj.com - GSS1
      ii. spoj.com - GSS2
      iii. codeforces.com - Classic Segment Tree (Expert Level)
      iv. spoj.com - IOPC1207
      v. spoj.com - ORDERSET
      vi. spoj.com - HELPR2D2
      vii. spoj.com - ANDROUND
      viii. spoj.com - HEAPULM
      ix. spoj.com - NICEDAY
      x. spoj.com - YODANESS
      xi. spoj.com - DQUERY
      xii. spoj.com - KQUERY
      xiii. spoj.com - FREQUENT
      xiv. spoj.com - GSS3
      xv. spoj.com - GSS4
      xvi. spoj.com - GSS5
      xvii. spoj.com - KGSS
      xviii. spoj.com - HELPR2D2
      xix. spoj.com - BRCKTS
      xx. spoj.com - CTRICK
      xxi. spoj.com - MATSUM
      xxii. spoj.com - RATING
      xxiii. spoj.com - RRSCHED
      xxiv. spoj.com - SUPPER
      xxv. spoj.com - ORDERS
      xxvi. codechef.com - LEBOBBLE
      xxvii. codechef.com - QUERY
      xxviii. spoj.com - TEMPLEQ
      xxix. spoj.com - DISUBSTR
      xxx. spoj.com - QTREE
      xxxi. spoj.com - QTREE2
      xxxii. spoj.com - QTREE3
      xxxiii. spoj.com - QTREE4
      xxxiv. spoj.com - QTREE5
   c. Problems on segment tree with lazy propagation

Overview
Updates
Levels
Price & Details
Schedule
Prepare
Foundation
Advanced
Expert
Examination
Certified Candidates New
FAQ
Contact Us
Be a Representative
Dashboard

iii. spoj.com - MULTQ3 (another nice lazy propagation problem)
iv. codechef.com - CHEFD
v. codechef.com - FUNAGP (a difficult lazy propagation problem.)
vi. RPAR (a difficult and nice lazy propagation)
vii. codechef.com - ADDMUL
viii. spoj.com - SEGSQRSS (a difficult lazy propagation problem)
ix. spoj.com - KGSS
x. codeforces.com - C. Circular RMQ
xi. codeforces.com - E. Lucky Queries (must do hard problem on lazy propagation)
xii. codeforces.com - E. A Simple Task
xiii. codeforces.com - C. DZY Loves Fibonacci Numbers (important problem to do, introduces some nice properties over lazy propagation)
xiv. codeforces.com - D. The Child and Sequence
xv. codeforces.com - E. Lucky Array

Overview
Updates
Levels
Price & Details
Schedule
Prepare
Foundation
Advanced
Expert
Examination
Certified Candidates New
FAQ
Contact Us
Be a Representative
Dashboard

4. **Binary Index Tree (Fenwick tree)**
   a. Resources
      i. topcoder.com
      ii. iarcs.org.in
      iii. visualgo.net
   b. Practice Problems:
      Please solve the problems mentioned in the above segment tree practice problems section. Note that usually, it's difficult to do range updates in binary indexed trees. Mostly, it is used for for range query and point update. However, you can check the following article for checking how some simple specific kind of range updates can be peformed on binary indexed tree (http://petr-mitrichev.blogspot.in/2013/05/fenwick-tree-range-updates.html). Note that range updates on BIT is not a part of the syllabus.
      i. spoj.com - INVCNT
      ii. spoj.com - TRIPINV

5. **Trees (traversals)**
   a. Resources
      i. slideshare.net
      ii. iarcs.org.in
      iii. berkeley.edu
   b. Practice Problems
      i. spoj.com - TREEORD

6. **Finding Lowest Common Ancestors (O(log N) solution where N is number of nodes)**
   a. Resources
      i. topcoder.com

7. **Depth First Search, Breadth First Search (Finding connected components and transitive closures)**
   a. Resources
      i. geeksforgeeks.org - Connected Components in an undirected graph
      ii. geeksforgeeks.org - Transitive closure of a graph
      iii. geeksforgeeks.org - Depth First Traversal or DFS for a Graph
      iv. iarcs.org.in - Basic Graph Algorithms
      v. visualgo.net - Graph Traversal
      vi. harvard.edu - Breadth-First Search
   b. Practice Problems
      i. codechef.com - FIRESC, editorial
      ii. spoj.com - BUGLIFE
      iii. spoj.com - CAM5

vii. spoj.com - PT07Z
viii. spoj.com - LABYR1
ix. spoj.com - PARADOX
x. spoj.com - PPATH ;(must do bfs problem)
xi. spoj.com - ELEVTRBL (bfs)
xii. spoj.com - QUEEN (bfs)
xiii. spoj.com - SSORT ;(cycles in a graph)
xiv. spoj.com - ROBOTGRI ;(bfs)

8. **Shortest-path algorithms (Dijkstra, Bellman-Ford, Floyd-Warshall)**
   a. Resources
      i. geeksforgeeks.org - Dijkstra's shortest path algorithm
      ii. iarcs.org.in - Shortest paths
      iii. Visualgo.net - Single-Source Shortest Paths (SSSP)
   b. Practice Problems
      i. codechef.com - DIGJUMP, editorial
      ii. codechef.com - AMR14B, editorial
      iii. codechef.com - INSQ15_F, editorial
      iv. codechef.com - SPSHORT, editorial (slightly difficult dijkstra's problem.)
      v. codechef.com - RIVPILE, editorial
      vi. spoj.com - SHPATH
      vii. spoj.com - TRAFFICN
      viii. spoj.com - SAMER08A
      ix. spoj.com - MICEMAZE
      x. spoj.com - TRVCOST
      xi. codechef.com - PAIRCLST, editorial

9. **Bellman Ford Algorithm**
   a. Resources
      i. geeksforgeeks.org - Dynamic Programming - Bellman–Ford Algorithm
      ii. compprog.wordpress.com - ;One Source Shortest Path - Bellman-Ford Algorithm
   b. Practice Problem
      i. community.topcoder.com - PeopleYouMayKnow
      ii. codeforces.com - D. Robot Control
      iii. spoj.com - ARBITRAG - Arbitrage ;(Floyd Warshall)
      iv. community.topcoder.com - NetworkSecurity ;(Floyd Warshall)

10. **Minimum spanning tree (Prim and Kruskal algorithms)**
    a. Resources
       i. algs4.cs.princeton.edu - Minimum Spanning Trees
       ii. iarcs.org.in - Spanning trees
       iii. visualgo.net - Spanning Tree
    b. Practice Problem
       i. spoj.com - MST
       ii. spoj.com - NITTROAD
       iii. spoj.com - BLINNET
       iv. spoj.com - CSTREET
       v. spoj.com - HIGHWAYS
       vi. spoj.com - IITWPC4I
       vii. codechef.com - MSTQS, editorial
       viii. codechef.com - CHEFGAME, editorial
       ix. codechef.com - GALACTIK, editorial

Overview
Updates
Levels
Price & Details
Schedule
Prepare
Foundation
Advanced
Expert
Examination
Certified Candidates New
FAQ
Contact Us
Be a Representative
Dashboard

11. **Biconnectivity in undirected graphs (bridges, articulation points)**
    a. Resources
        i. e-maxx-eng.appspot.com - Finding Bridges in a Graph
        ii. iarcs.org.in - Articulation Points
        iii. pisces.ck.tp.edu.tw - Articulation Points
    b. Practice Problem
        i. uva.onlinejudge.org - Network
        ii. icpcarchive.ecs.baylor.edu - Building Bridges
        iii. uva.onlinejudge.org - Tourist Guide
        iv. acm.tju.edu.cn - Network
        v. spoj.com - EC_P - Critical Edges
        vi. spoj.com - SUBMERGE - Submerging Islands
        vii. spoj.com - POLQUERY - Police Query
        viii. codeforces.com - A. Cutting Figure
12. **Strongly connected components in directed graphs**
    a. Resources
        i. iarcs.org.in - Strongly connected components
        ii. theory.stanford.edu - Strongly Connected Components
    b. Practice Problem
        i. spoj.com - ANTTT
        ii. spoj.com - CAPCITY
        iii. spoj.com - SUBMERGE
        iv. codechef.com - MCO16405, editorial
        v. spoj.com - BOTTOM
        vi. spoj.com - BREAK
        vii. community.topcoder.com - Marble Collection Game
13. **Topological Sorting**
    a. Resources
        i. geeksforgeeks.org - Topological Sorting
    b. Practice Problem
        i. spoj.com - TOPOSORT ;
        ii. codeforces.com - C. Fox And Names ;
        iii. codechef.com - RRDAG, editorial
        iv. spoj.com - RPLA
        v. codechef.com - CL16BF (topological sort with dp), editorial
        vi. spoj.com - MAKETREE
14. **Euler path, tour/cycle.**
    a. Resources
        i. math.ku.edu - Euler Paths and Euler Circuits
    b. Practice Problem
        i. spoj.com - WORDS1
        ii. codechef.com - CHEFPASS, editorial
        iii. codechef.com - TOURISTS, editorial
        iv. codeforces.com - D. New Year Santa Network
        v. B. Strongly Connected City
        vi. codechef.com - PEOPLOVE
        vii. codeforces.com - D. Tanya and Password
        viii. codeforces.com - E. One-Way Reform
        ix. spoj.com - GCPC11C

Overview
Updates
Levels
Price & Details
Schedule
Prepare
Foundation
Advanced
Expert
Examination
Certified Candidates New
FAQ
Contact Us
Be a Representative
Dashboard

- Overview
- Updates
- Levels
- Price & Details
- Schedule
- **Prepare**
  - Foundation
  - Advanced
  - **Expert**
- Examination
- Certified Candidates New
- FAQ
- Contact Us
- Be a Representative
- Dashboard

a. Resources

i. codechef.com - Fast Modulo Multiplication (Exponential Squaring)

ii. codechef.com - Best known algos for calculating nCr % M ;(only for expert level)

16. **Amortized Analysis**

a. Resources

i. ocw.mit.edu - Amortized Analysis

ii. wikipedia.org - Amortized Analysis

iii. iiitdm.ac.in - Amortized Analysis

17. **Divide and Conquer**

a. Resources

i. cs.cmu.edu - Divide-and-Conquer and Recurrences

ii. geeksforgeeks.org - Divide-and-Conquer

b. Practice Problem

i. codechef.com - MRGSRT, editorial

ii. spoj.com - HISTOGRA

iii. codechef.com - TASTYD, editorial

iv. codechef.com - RESTPERM, editorial

v. codechef.com - ACM14KP1, editorial

18. **Advanced Dynamic Programming** problems (excluding the dp optimizations which are added in expert level, Please go through the basic DP resources and problems mentioned in foundation level resource.)

a. Resources

i. apps.topcoder.com - Commonly used DP state domains

ii. apps.topcoder.com - Introducing Dynamic Programming

iii. apps.topcoder.com - Optimizing DP solution

iv. codeforces.com - DP over Subsets and Paths

b. Problems for Advanced DP

i. spoj.com - HIST2 ;(dp bitmask)

ii. spoj.com - LAZYCOWS ;(dp bitmask)

iii. spoj.com - TRSTAGE ;(dp bitmask)

iv. spoj.com - MARTIAN

v. spoj.com - SQRBR

vi. spoj.com - ACMAKER

vii. spoj.com - AEROLITE

viii. spoj.com - BACKPACK

ix. spoj.com - COURIER

x. spoj.com - DP

xi. spoj.com - EDIST

xii. spoj.com - KRECT

xiii. spoj.com - GNY07H

xiv. spoj.com - LISA

xv. spoj.com - MINUS

xvi. spoj.com - NAJKRACI

xvii. spoj.com - PHIDIAS

xviii. spoj.com - PIGBANK

xix. spoj.com - PT07X

xx. spoj.com - VOCV

xxi. spoj.com - TOURIST

xxii. spoj.com - MKBUDGET

xxiii. spoj.com - MMAXPER

xxvi. spoj.com - RENT ;(dp with segment tree/BIT)
xxvii. spoj.com - INCSEQ ;(dp with segment tree/BIT)
xxviii. spoj.com - INCDSEQ ;(dp with segment tree/BIT)
xxix. You can solve some advanced problems from
xxx. codeforces.com - Dynamic Programming Type

19. **Sieve of Eratosthenes**
    a. Resources:
        i. codechef.com - Sieve Methods
    b. Practice Problems
        i. spoj.com - TDKPRIME
        ii. spoj.com - TDPRIMES
        iii. spoj.com - ODDDIV ;(sieve + binary search)
        iv. spoj.com - NDIVPHI ;O(N) prime testing algorithm)
        v. spoj.com - DIV ;(divisor sieve)
        vi. codechef.com - LEVY, editorial
        vii. codechef.com - PRETNUM, editorial
        viii. codechef.com - KPRIME, editorial
        ix. codechef.com - DIVMAC, editorial (segment tree with sieve)
        x. codechef.com - PPERM, editorial ;(a bit advanced sieve application)
20. **General**
    a. Stanford Algoriths 1
    b. Stanford Algoriths 2

Mock Test:

1. Test 1 - codechef.com/ALMOCK01
2. Test 2 - codechef.com/ALMOCK02
3. Test 3 - codechef.com/ALMOCK03
4. Test 4 - codechef.com/ALMOCK04

Overview
Updates
Levels
Price & Details
Schedule
Prepare
Foundation
Advanced
Expert
Examination
Certified Candidates New
FAQ
Contact Us
Be a Representative
Dashboard

Expert

This level is intended to test that the candidate is an expert in algorithms and data structures, and has a deep understanding of the topics. Candidates can expect problems from the following topics to come in the exam.

Syllabus:

The syllabus for Expert Level is open-ended. Everything in Advanced Level will be included, along with:

1. Treaps
2. Persistent Data Structures
3. HLD
4. Centroid Decomposition
5. Computational Geometry
6. Fast Fourier Transforms
7. Game Theory
8. Gaussian Elimination
9. Dynamic Programming Optimizations (eg. Convex Hull Trick, Divide and Conquer Optimization, Knuth Optimization)
10. Advanced String algorithms (Tries, KMP, Aho-Corasik, Suffix arrays, Suffix trees)
11. Flows (Max-Flow, Min Cost Max Flow)

1. The resources are listed here.

Mock Test:

1. Test 1 - codechef.com/ELMOCK01
2. Test 2 - codechef.com/ELMOCK02
3. Test 3 - codechef.com/ELMOCK03
4. Test 4 - codechef.com/ELMOCK04

Note: These links have been curated to help in preparation for the exams, and also to help the community in general. But if you own some of the material linked to, and you wouldn't like them to be here, please contact us, and we will remove it.

▸ Overview
Updates

CodeChef - A Platform for Aspiring Programmers
CodeChef was created as a platform to help programmers make it big in the world of algorithms, **computer programming** and **programming contests**. At CodeChef we work hard to revive the geek in you by hosting a **programming contest** at the start of the month and another smaller programming challenge in the middle of the month. We also aim to have training sessions and discussions related to **algorithms**, **binary search**, technicalities like **array size** and the likes. Apart from providing a platform for **programming competitions**, CodeChef also has various algorithm tutorials and forum discussions to help those who are new to the world of **computer programming**.

Practice Section - A Place to hone your 'Computer Programming Skills'
Try your hand at one of our many practice problems and submit your solution in a language of your choice. Our **programming contest** judge accepts solutions in over 35+ programming languages. Preparing for coding contests were never this much fun! Receive points, and move up through the CodeChef ranks. Use our practice section to better prepare yourself for the multiple **programming challenges** that take place through-out the month on CodeChef.

Compete - Monthly Programming Contests and Cook-offs
Here is where you can show off your **computer programming skills**. Take part in our 10 day long monthly coding contest and the shorter format Cook-off **coding contest**. Put yourself up for recognition and win great prizes. Our **programming contests** have prizes worth up to INR 20,000 (for Indian Community), $700 (for Global Community) and lots more CodeChef goodies up for grabs.