SPOJ QUESTIONS LIST

List compiled by Anmol Deep (IT 2k14)

Beginners can start from the below questions on spoj and then move towards more difficult problems. Please try to solve these questions yourself without referring to the solution.

ADDREV
EC_CONB
CRDS
ACPC11B
CODCHESS
FASHION

Before moving to the list, solve these basic adhoc questions for best results.

ARMY
ESYRCRTN
FCTRL
FCTRL2
IEEEBGAM
PHT
SPCQ
SPCU
MAY99_2
MAY99_3
ENIGMATH
CEQU
MKEQUAL
SNGPG
SAMER08F
WILLITST
MOHIB
HANGOVER
CANDY
CANDY3
NSTEPS
SILVER
KURUK14
NITK06

After solving the above questions you can solve below questions.

Contents:-

About This:-

This is a very comprehensive list, solving this will get about 90% of your preparation done. Questions in each set are sorted according to their difficulty, but
you can always try the next question if you get stuck. Everything from Math to DFS/BFS is very important, I recommend solving all question in these topics as they
are mostly of easy or medium difficulty and will teach you a LOT of things. ADHOC is tougher compared to the last set, but there are plenty of alternatives. Nevertheless, these will definitely improve implementation skills. There are some basic questions on DSU. There is a section on backtracking which will cover Josephus
algorithm.

The next set of questions will cover basic Dynamic Programming, MST, SCC, Shortest Path algorithms, more problems on binary search, data structures, graph theory, dsu
AND string algorithms like KMP.

All the best!
****

Somewhat tougher than the last implementation questions. Some may take lot of time but worth the effort.

ADHOC:-

BUSYMAN
GERGOVIA
KNJIGE
CUBARTWK
VAPI01
SNGMSG
PWRARR
MAIN12A
PQUEUE
CATM
UOFTAB
JAVAC
PALIN

QUE1

Math:-

DOL
MOHIB
ABSP1
QUADAREA
GIRLSNBS
EBOXES

Learn Modular Exponentiation & Modulo Inverse (Very Important Topics, Used In Lots Of Problems, However Could Not Remember Most Of Them)

ZSUM
RIVALS
ADST01

Learn Euclidean GCD

SPEED
STREETR
CEQU
GCD2

Learn Optimized Sieve

TDPRIMES
TDKPRIMES
CUBEFR
MCUR98
HARSHAD

Learn Horner's method

POLEVAL

Learn Euler’s Totient Function

ETF
STARSBC
FACT0

Binary Search:-

HACKRNDM
EKO
MAIN8_C
EGYPIZZA
NOTATRI
AGGRCOW
CISTFILL
CURDPROD
Do read Topcoder Tutorial on binary search before attempting these questions. They cover all the possible edge cases.

C++ STL and Basic Data Structures(Stack, Queue etc):-

RPLE
STPAR
ANARC09A ****(Tagged under DP, alternative solution exists)
FACEFRND
MRECAMAN
MAJOR
PRO
RKS
SBANK
HOMO
ASCDFIB
BOI7SEQ ****Optional. Hardest in this set.

Sliding Window/ Two Pointers:-

ALIEN
ARRAYSUB
HOTELS
BOI7SOU

DFS/BFS:-

CAM5
BUGLIFE
NAKANJ
PPATH
ELEVTRBL
PT07Y
PT07Z
PYRA
AKBAR

DFS/BFS on 2D grid:-

ABCPATH
BITMAP
UCV2013H

DSU:-

SOCNETC
FRNDCIRC
LOSTNSURVIVED
FOXLINGS

BACKTRACKING:-

Learn Josephus
DANGER
WTK
POCRI
*****
NG0FRCTN ****Perhaps hardest among all these questions. Optional.

CONTENTS
1.> Dynamic Programming
2.> Graph Algorithms(SCC+Topological Sort+Articulation Points+Lowest Common Ancestor+DFS/BFS)
3.> MST & Dijkstra
4.> DSU
5.> KMP/String Algorithms
6.> Segment Tree/Binary Index Tree
7.> Greedy/Adhoc/Math/Binary Search

This list is somewhat less comprehensive as questions from such algorithms are hard to find and even harder to solve. It is still a great collection for getting started
on SPOJ. The section in DP contains some classic techniques which need to be studied beforehand.

****

Dynamic Programming:

1.> FARIDA
2.> ALIEN2
3.> DCEPC501
4.> ACPC10D
5.> ACODE
6.> WACHOVIA (Knapsack)
7.> TRT
8.> TWENDS
9.> NFURY
10.> NY10E
11.> MAXWOODS (Min Cost Path)
12.> ELIS (Longest Increasing Subsequence)
13.> EDIST (Edit Distance)
14.> EDIT
15.> MAY99_4 (Binomial Coefficient)
16.> GOO
17.> CRSCNTRY (Longest Common Subsequence)
18.> AIBOHP
19.> MMAXPER
20.> MCOINS

21.> COINS
22.> PARTY
23.> PIGBANK
24.> MINVEST
25.> SCUBADIV
26> RPLB
27.> NOCHANGE
28.> FPOLICE
29.> CHOCOLA
30.> BAT3
31.> ALTSEQ
32.> SMILEY1807
33.> PHIDIAS
34.> BABTWR
35.> RENT
36.> ORDSUM23
37.> CZ_PROB1
38.> UOFTAE
39.> PPBRJB
40.> ROCK
41.> SAFECRAC
42.> SAMER08C
43.> MAIN72 (Subset Sum)
44.> MAIN113
45.> PERMUT1
46.> PT07X (Vertex Cover)
47.> LPIS
48.> MKBUDGET
49.> PERMUT1
50.> LOVEBIRDS
51.> TEMPTISL
52.> PRUBALL (Egg Dropping Puzzle)
53.> MIXTURES (Matrix Chain Multiplication)
54.> LISA
55.> CODERE3 (Longest Bitonic Subsequence)
56.> MARTIAN
57.> DSUBSEQ
58.> BVAAN

This does not cover all dp topics from geeksforgeeks such as the cutting rod problem,box stacking problem etc, but will still provide a good foundation on dynamic programming.

All the best!

****

GRAPH ALGORITHMS:-

ADVANCED DFS/BFS AND MISC GRAPH THEORY:-

1.> MLASERP
2.> ESJAIL
3.> ESCJAILA
4.> ONEZERO
5.> MOHIBTREE
6.> CFPARTY
7.> ANARC08G
8.> PARADOX
9.> HERDING

MST/DIJKSTRA & SHORTEST PATHS:-

1.> SHPATH
2.> ULM09
3.> BLINNET
4.> BENEFACT
5.> CHICAGO
6.> IITWPC4I
7.> MARYBMW
8.> INCARDS
9.> TRAFFICN
10.> SAMER08A
11.> KOICOST

SCC (Lowest Common Ancestor + Topological Sort + Articulation Points):-

1.> TOUR
2.> BOTTOM
3.> CAPCITY
4.> WEBISL
5.> LCA
6.> SUBMERGE (ARTICULATION POINTS)
7.> TOPOSORT (TOPOLOGICAL SORT) **********BTW USE KAHN'S INDEGREE METHOD INSTEAD OF TARJAN FOR TOPOLOGICAL SORTING, SIMPLER TO IMPLEMENT
8.> PFDEP
9.> EC_P

DSU :-

1.> BTCODE_G
2.> CORNET
3.> LOSTNSURVIVED
4.> FOXLINGS (CO-ORDINATE COMPRESSION)

KMP/STRING ALGORITHMS:-

NHAY
FILRTEST
TESSER
EPALIN

PERIOD

SEGMENT TREE/BINARY INDEXED TREE:-

1.> AKVQLD03
2.> ANDROUND
3.> INVCNT
4.> HORRIBLE (Lazy Propagation)
5.> LITE
6.> MULTQ3
7.> RMID
8.> RPLN
9.> RATINGS
10.> DCEPC206
11.> INCSEQ

MORE PROBLEMS ON GREEDY/MATH/BINARY SEARCH:-

1.> ABCDEF
2.> SUBS
3.> SUBSUMS (MEET IN THE MIDDLE)
4.> NR2
5.> ARRANGE
6.> SECTORS
7.> POTIONS
8.> GCDEX
9.> IITKWPCN