

# **Coding Interview Questions**

**By  
Narasimha Karumanchi**

 **Concepts**    **Problems**    **Interview Questions**

Copyright ©2015 by *CareerMonk.com*

All rights reserved.

Designed by *Narasimha Karumanchi*

Copyright ©2015 CareerMonk Publications. All rights reserved.

All rights reserved. No part of this book may be reproduced in any form or by any electronic or mechanical means, including information storage and retrieval systems, without written permission from the publisher or author

# Acknowledgements

*Mother and father*, it is impossible to thank you adequately for everything you have done, from loving me unconditionally to raising me in a stable household, where you persistent efforts traditional values and taught your children to celebrate and embrace life. I could not have asked for better parents or role-models. You showed me that anything is possible with faith, hard work and determination.

This book would not have been possible without the help of many people. I would like to thank them for their efforts in improving the end result. Before we do so, however, I should mention that I have done my best to correct the mistakes that the reviewers have pointed out and to accurately describe the protocols and mechanisms. I alone am responsible for any remaining errors.

First and foremost, I would like to express my gratitude to many people who saw me through this book, to all those who provided support, talked things over, read, wrote, offered comments, allowed me to quote their remarks and assisted in the editing, proofreading and design. In particular, I would like to thank the following individuals.

- *Mohan Mullapudi*, IIT Bombay, Architect, dataRPM Pvt. Ltd.
- *Navin Kumar Jaiswal*, Senior Consultant, Juniper Networks Inc.
- *A.Vamshi Krishna*, IIT Kanpur, Mentor Graphics Inc.
- *Hirak Chatterjee*, Yahoo Inc.
- *Kondrakunta Murali Krishna*, B-Tech., Technical Lead, HCL
- *Chaganti Siva Rama Krishna Prasad*, Founder, StockMonks Pvt. Ltd.
- *Naveen Valsakumar*, Co-Founder, NotionPress Pvt. Ltd.
- *Ramanaiah*, Lecturer, Nagarjuna Institute of Technology and Sciences, MLG

Last but not least, I would like to thank *Directors of Guntur Vikas College*, *Prof.Y.V.Gopala Krishna Murthy & Prof.Ayub Khan* [ACE Engineering Academy], *T.R.C.Bose* [Ex. Director of APTransco], *Ch.Venkateswara Rao VNR Vignanjyothi* [Engineering College, Hyderabad], *Ch.Venkata Narasaiah* [IPS], *Yarapathineni Lakshmaiah* [Manchikallu, Gurazala] and all our well – wishers for helping me and my family during our studies.

-Narasimha Karumanchi  
M-Tech, IIT Bombay  
Founder, CareerMonk.com



# Preface

Dear Reader,

**Please Hold on!** I know many people do not read the preface. But I would strongly recommend that you go through the preface of this book at least. The reason for this is that this preface has *something different* to offer.

This book assumes you have some basic knowledge about computer science. The main objective of the book is not to give you the theorems and proofs about *Data Structures* and *Algorithms*. I have followed a pattern of improving the problem solutions with different complexities (for each problem, you will find multiple solutions with different, and reduced complexities). Basically, it's an enumeration of possible solutions. With this approach, even if you get a new question it will show you a way to think about all possible solutions. This book is very useful for interview preparation, competitive exams preparation, and campus interview preparations.

As a *job seeker* if you read the complete book with good understanding, I am sure you will challenge the interviewers and that is the objective of this book.

This book is very useful for the *students of Engineering Degree and Masters* during their academic preparations. In all the chapters you will see that more importance has been given to problems and their analysis instead of theory. For each chapter, first you will read about the basic required theory and this will be followed by a section on problem sets. There are approximately 700 algorithmic problems and all of them are with solutions.

In most the chapters you will see more importance given to *problems* and analyzing them instead of concentrating more on theory. For each chapter, first you will see the basic required theory and then followed by problems.

For many problems, *multiple* solutions are provided with different levels of complexities. We start with *brute force* solution and slowly move towards the *best solution* possible for that problem. For each problem we will try to understand how much time the algorithm is taking and how much memory the algorithm is taking.

It is *recommended* that the reader does at least one complete reading of this book to get full understanding of all the topics. In the subsequent readings, you can go directly to any chapter and refer. Even though, enough readings were given for correcting the errors, there could be some minor typos in the book. If any such typos are found, they will be updated at [www.CareerMonk.com](http://www.CareerMonk.com). I request you to constantly monitor this site for any corrections, new problems and solutions. Also, please provide your valuable suggestions at: [Info@CareerMonk.com](mailto:Info@CareerMonk.com).

Wish you all the best. I am sure that you will find this book useful.

-Narasimha Karumanchi  
M-Tech, IIT Bombay  
Founder, [CareerMonk.com](http://CareerMonk.com)



# Table of Contents

1. Programming Basics .....	13
1.1 Variables-----	13
1.2 Data types-----	13
1.3 Data Structure-----	13
1.4 Abstract Data Types (ADTs)-----	14
1.5 Memory and Variables-----	14
1.6 Pointers-----	15
1.7 Techniques of Parameter Passing-----	17
1.8 Binding-----	20
1.9 Scope-----	20
1.10 Storage Classes-----	21
1.11 Storage Organization-----	25
1.12 Programming Techniques-----	27
1.13 Basic Concepts of OOPS-----	28
2. Scripting Languages .....	71
2.1 Interpreter versus Compiler-----	71
2.2 What Are Scripting Languages?-----	72
2.3 Shell Scripting-----	72
2.4 PERL [Practical Extraction and Report Language]-----	78
2.5 Python-----	95
3. Design Interview Questions .....	100
3.1 Glossary-----	100
3.2 Tips-----	101
3.3 Sample Design Questions For Practice-----	142
4. Operating System Concepts .....	143
4.1 Glossary-----	143
4.2 Questions on Operating System Concepts-----	144
5. Computer Networking Basics .....	148
5.1 Introduction-----	148
5.2 LAN vs. WAN-----	148
5.2 Segmentation and Multiplexing-----	149
5.3 End Devices-----	149
5.4 Intermediary Devices-----	149
5.5 Hub, Switch, and Router Defined-----	149
5.6 Medium-----	150
5.7 Peer-to-peer and Client/server networks-----	150
5.8 How does Internet works?-----	151
5.9 Difference between OSI and TCP/IP models-----	153
5.10 Client/Server Computing and the Internet-----	154
5.11 ARP and RARP-----	154
5.12 Subnetting-----	155
5.13 How Routing Works?-----	155
5.14 Unicast, Broadcast and Multicast-----	156
5.15 How traceroute (or tracert) and ping works?-----	156
5.16 What is QoS?-----	157
6. Database Concepts.....	158
6.1 Glossary-----	158
6.2 Questions on Database Concepts-----	159
7. Brain Teasers .....	163
7.1 Questions on Brain Teasers-----	163
8. Algorithms Introduction.....	165
8.1 What is an Algorithm?-----	165
8.2 Why Analysis of Algorithms?-----	165
8.3 Goal of Analysis of Algorithms-----	165

8.4 What is Running Time Analysis? .....	165
8.5 How to Compare Algorithms? .....	165
8.6 What is Rate of Growth? .....	166
8.7 Commonly used Rate of Growths .....	166
8.8 Types of Analysis .....	166
8.9 Asymptotic Notation.....	167
8.10 Big-O Notation.....	167
8.15 Omega- $\Omega$ Notation .....	169
8.16 Theta- $\Theta$ Notation .....	169
8.17 Why is it called Asymptotic Analysis? .....	170
8.18 Guidelines for Asymptotic Analysis .....	170
8.19 Properties of Notations .....	171
8.20 Commonly used Logarithms and Summations .....	171
8.21 Master Theorem for Divide and Conquer .....	172
8.22 Problems on Divide and Conquer Master Theorem .....	172
8.23 Master Theorem for Subtract and Conquer Recurrences .....	173
8.24 Variant of subtraction and conquer master theorem .....	173
8.25 Problems on Algorithms Analysis .....	173
<b>9. Recursion and Backtracking.....</b>	<b>182</b>
9.1 Introduction .....	182
9.2 What is Recursion?.....	182
9.3 Why Recursion?.....	182
9.4 Format of a Recursive Function.....	182
9.5 Recursion and Memory (Visualization).....	183
9.6 Recursion versus Iteration .....	183
9.7 Notes on Recursion.....	183
9.8 Example Algorithms of Recursion .....	184
9.9 Problems on Recursion .....	184
9.10 What is Backtracking? .....	185
9.11 Example Algorithms of Backtracking.....	185
9.12 Problems on Backtracking .....	185
<b>10. Linked Lists .....</b>	<b>188</b>
10.1 What is a Linked List? .....	188
10.2 Linked Lists ADT.....	188
10.3 Why Linked Lists? .....	188
10.4 Arrays Overview .....	188
10.5 Comparison of Linked Lists with Arrays & Dynamic Arrays.....	189
10.6 Singly Linked Lists.....	189
10.7 Doubly Linked Lists .....	194
10.8 Circular Linked Lists .....	198
10.9 Memory-Efficient Doubly Linked List.....	203
10.10 Unrolled Linked Lists .....	204
10.11 Skip Lists .....	208
10.12 Problems on Linked Lists .....	212
<b>11. Stacks.....</b>	<b>229</b>
11.1 What is a Stack? .....	229
11.2 How Stacks are used?.....	229
11.3 Stack ADT.....	229
11.4 Applications.....	230
11.5 Implementation.....	230
11.6 Comparison of Implementations .....	234
11.7 Problems on Stacks.....	234
<b>12. Queues .....</b>	<b>250</b>
12.1 What is a Queue? .....	250
12.2 How are Queues Used? .....	250
12.3 Queue ADT.....	250
12.4 Exceptions .....	251
12.5 Applications.....	251
12.6 Implementation.....	251
12.7 Problems on Queues .....	256
<b>13. Trees.....</b>	<b>261</b>
13.1 What is a Tree?.....	261
13.2 Glossary .....	261



13.3 Binary Trees	262
13.4 Types of Binary Trees	262
13.5 Properties of Binary Trees	263
13.6 Binary Tree Traversals	264
13.7 Generic Trees (N-ary Trees)	282
13.8 Threaded Binary Tree [Stack/Queue less] Traversals	287
13.9 Expression Trees	292
13.10 XOR Trees	294
13.11 Binary Search Trees (BSTs)	295
13.12 Balanced Binary Search Trees	307
13.13 AVL (Adelson-Velskii and Landis) Trees	308
13.14 Other Variations in Trees	321
14. Priority Queue and Heaps	324
14.1 What is a Priority Queue?	324
14.2 Priority Queue ADT	324
14.3 Priority Queue Applications	324
14.4 Priority Queue Implementations	324
14.5 Heaps and Binary Heap	325
14.6 Binary Heaps	326
14.7 Problems on Priority Queues [Heaps]	332
15. Graph Algorithms	341
15.1 Introduction	341
15.2 Glossary	341
15.3 Applications of Graphs	343
15.4 Graph Representation	344
15.5 Graph Traversals	346
15.6 Topological Sort	352
15.7 Shortest Path Algorithms	353
15.8 Minimal Spanning Tree	358
15.9 Problems on Graph Algorithms	361
16. Sorting	366
16.1 What is Sorting?	366
16.2 Why is Sorting necessary?	366
16.3 Classification of Sorting Algorithms	366
16.4 Other Classifications	366
16.5 Bubble sort	367
16.6 Selection Sort	367
16.7 Insertion sort	368
16.8 Shell sort	370
16.9 Merge sort	371
16.10 Heapsort	372
16.11 Quicksort	372
16.12 Tree Sort	374
16.13 Comparison of Sorting Algorithms	374
16.14 Linear Sorting Algorithms	375
16.15 Counting Sort	375
16.16 Bucket sort [or Bin Sort]	375
16.17 Radix sort	376
16.18 Topological Sort	376
16.19 External Sorting	376
16.20 Problems on Sorting	377
17. Searching	382
17.1 What is Searching?	382
17.2 Why do we need Searching?	382
17.3 Types of Searching	382
17.4 Unordered Linear Search	382
17.5 Sorted/Ordered Linear Search	382
17.6 Binary Search	383
17.7 Comparing Basic Searching Algorithms	383
17.8 Symbol Tables and Hashing	383
17.9 String Searching Algorithms	383
17.10 Problems on Searching	384
18. Selection Algorithms [Medians]	403

18.1 What are Selection Algorithms? .....	403
18.2 Selection by Sorting .....	403
18.3 Partition-based Selection Algorithm .....	403
18.4 Linear Selection algorithm - Median of Medians algorithm .....	403
18.5 Finding the K Smallest Elements in Sorted Order .....	403
18.6 Problems on Selection Algorithms .....	403
19. Symbol Tables .....	411
19.1 Introduction .....	411
19.2 What are Symbol Tables? .....	411
19.3 Symbol Table Implementations .....	411
19.4 Comparison of Symbol Table Implementations .....	412
20. Hashing .....	413
20.1 What is Hashing? .....	413
20.2 Why Hashing? .....	413
20.3 HashTable ADT .....	413
20.4 Understanding Hashing .....	413
20.5 Components of Hashing .....	414
20.6 Hash Table .....	414
20.7 Hash Function .....	415
20.8 Load Factor .....	415
20.9 Collisions .....	415
20.10 Collision Resolution Techniques .....	415
20.11 Separate Chaining .....	415
20.12 Open Addressing .....	416
20.13 Comparison of Collision Resolution Techniques .....	417
20.14 How Hashing Gets $O(1)$ Complexity? .....	418
20.15 Hashing Techniques .....	418
20.16 Problems for which Hash Tables are not Suitable .....	418
20.7 Bloom Filters .....	418
20.18 Problems on Hashing .....	420
21. String Algorithms .....	428
21.1 Introduction .....	428
21.2 String Matching Algorithms .....	428
21.3 Brute Force Approach .....	428
21.4 Robin-Karp String Matching Algorithm .....	429
21.5 String Matching with Finite Automata .....	429
21.6 KMP Algorithm .....	430
21.7 Boyce-Moore Algorithm .....	433
21.8 Data Structures for Storing Strings .....	433
21.9 Hash Tables for Strings .....	433
21.10 Binary Search Trees for Strings .....	433
21.11 Tries .....	434
21.12 Ternary Search Trees .....	436
21.13 Comparing BSTs, Tries and TSTs .....	439
21.14 Suffix Trees .....	439
21.15 Problems on Strings .....	442
22. Algorithms Design Techniques .....	447
22.1 Introduction .....	447
22.2 Classification .....	447
22.3 Classification by Implementation Method .....	447
22.4 Classification by Design Method .....	448
22.5 Other Classifications .....	448
23. Greedy Algorithms .....	450
23.1 Introduction .....	450
23.2 Greedy strategy .....	450
23.3 Elements of Greedy Algorithms .....	450
23.4 Does Greedy Always Work? .....	450
23.5 Advantages and Disadvantages of Greedy Method .....	450
23.6 Greedy Applications .....	450
23.7 Understanding Greedy Technique .....	451
23.8 Problems on Greedy Algorithms .....	453
24. Divide and Conquer Algorithms .....	457

24.1 Introduction .....	457
24.2 What is Divide and Conquer Strategy? .....	457
24.3 Does Divide and Conquer Always Work? .....	457
24.4 Divide and Conquer Visualization .....	457
24.5 Understanding Divide and Conquer .....	458
24.6 Advantages of Divide and Conquer .....	458
24.7 Disadvantages of Divide and Conquer .....	458
24.8 Master Theorem .....	458
24.9 Divide and Conquer Applications .....	459
24.10 Problems on Divide and Conquer .....	459
25. Dynamic Programming .....	469
25.1 Introduction .....	469
25.2 What is Dynamic Programming Strategy? .....	469
25.3 Properties of Dynamic Programming Strategy .....	469
25.4 Can Dynamic Programming Solve All Problems? .....	469
25.5 Dynamic Programming Approaches .....	469
25.6 Examples of Dynamic Programming Algorithms .....	470
25.7 Understanding Dynamic Programming .....	470
25.8 Problems on Dynamic Programming .....	474
26. Miscellaneous Concepts.....	499
26.1 Introduction .....	499
26.2 Hacks on Bitwise Programming.....	499
26.3 Other Programming Questions.....	502
27. Non-Technical Help .....	508
27.1 Tips .....	508
27.2 Sample Non-Technical Questions .....	509

# Coding Interview Questions

Other Titles by *Narasimha Karumanchi*

- 📖 IT Interview Questions
- 📖 Elements Of Computer Networking
- 📖 Data Structures And Algorithms Made Easy (C/C++)
- 📖 Data Structures And Algorithms Made Easy In Java
- 📖 Data Structure And Algorithmic Thinking With Python
- 📖 Data Structures And Algorithms For GATE
- 📖 Peeling Design Patterns