

# Training-Data Structure-Algorithm

**Dot Net & Java** 

Version 1.0

Abstract

DS Training Pathway



### **Contents**

1	Intr	oduction to document	E
	1.1	Intended audience	Œ
		-requisite	
		urse Material	
		Data Structure & Algorithm	
	3.1.	1 References	E
	3.2	Assignment	. 6
	3 3	Evaluation Criteria	-



## **Copyright Notice**

Copyright © 2021 Nagarro Software Pvt. Ltd. All rights reserved.

This document is confidential and proprietary to Nagarro Software Pvt. Ltd. and no part of it should be reproduced, published, transmitted or distributed in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, or stored in any information storage or retrieval system of any nature nor should it be disclosed to third parties without the prior express written authorization of Nagarro Software Pvt. Ltd.



Version 1.0 2 / 8

# 1 Introduction to document

### 1.1 Intended audience

The intended target group for this document are:

- Trainee
- Junior Associate



Version 1.0 3/8

# 2 Pre-requisite

Prior knowledge of C#/Java language.



Version 1.0 4 / 8

## 3 Course Material

## 3.1 Data Structure & Algorithm

Topic	Link	Hours
Common (Java & C#)	https://app.pluralsight.com/library/courses/algorithmics- introduction/transcript	6 hr.
Common (Java & C#) and C# example	<ul> <li>https://app.pluralsight.com/library/courses/algorithms-data-structures-part-one/transcript</li> <li>https://app.pluralsight.com/library/courses/algorithms-data-structures-part-two/transcript</li> </ul>	7 hr.
DS with Java examples (only for Java)	https://app.pluralsight.com/library/courses/java-data- structures-implementing-understanding/transcript	5 hr.

#### 3.1.1 References

Topic	Link
Book	<ul> <li>https://edutechlearners.com/download/Introduction_to_al gorithms-3rd%20Edition.pdf</li> </ul>
C# Samples	<ul> <li>https://www.youtube.com/watch?v=J9dqgrOSR2M</li> <li>https://www.youtube.com/watch?v=K0-qsnaUo</li> <li>https://www.youtube.com/watch?v=iKDhgVoXVTk</li> </ul>



Version 1.0 5 / 8

# 3.2 Assignment

S.no.	Assignment
1	Implement your own version of the Linked list with mentioned functionalities:  Insert Insert at position Delete Delete at position Center Reverse Size Iterator Traverse/Print
2	Implement your own version of the stack data with mentioned functionalities:  Push Pop Peek Contains Size Reverse Iterator Traverse/Print
3	Implement your own version of the following Queue with mentioned functionalities:  • Enqueue • Dequeue • Peek • Contains • Size • Reverse • Iterator • Traverse/Print
4	Implement your own version of the following Priority Queue with mentioned functionalities:  • Enqueue • Dequeue (Highest Priority) • Peek (Highest Priority) • Contains • Size • Reverse • Iterator



Version 1.0 6 / 8

	Traverse/Print
5 Implement your own version of the N-child tree with mentioned functionali	
	<ul> <li>Insert</li> <li>Delete</li> <li>Contains</li> <li>Get Elements by value</li> <li>Get Elements by level</li> <li>Iterator Breadth First</li> <li>Iterator Depth First</li> <li>Traverse/Print Breadth First</li> <li>Traverse/Print Depth First</li> </ul>
6	Implement your own version of the following Hash table with mentioned functionalities:  Insert Delete Contains Get Value by key Size Iterator Traverse/Print

## 3.3 Evaluation Criteria

Section	Points
Expected Behaviour and Output	<ul> <li>Use of similar data structures already present in the language/framework is not allowed.</li> <li>Make use of object-oriented capabilities for implementing the data structures.</li> <li>Exception handling is expected in the program</li> </ul>
Evaluation Criteria	<ul> <li>Code Completeness and Correctness</li> <li>Usage of OO Principles, package/class structure, class/function/variable names</li> <li>Complexity of the operations</li> <li>Code should be in running condition</li> <li>Presentation Skills</li> </ul>



Version 1.0 7 / 8