nagarro

Training-Data Structure-Algorithm

Dot Net & Java

Version 1.0

Abstract

DS Training Pathway

Contents

1 Introduction to document 3
1.1 Intended audience 3
2 Pre-requisite 4
3 Course Material 5
3.1 Data Structure & Algorithm 5
3.1.1 References 5
3.2 Assignment 6
3.3 Evaluation Criteria 7

Copyright Notice

1 Introduction to document

1.1 Intended audience

The intended target group for this document are:

- Trainee
- Junior Associate

2 Pre-requisite

Prior knowledge of C#/Java language.

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	• https://app.pluralsight.com/library/courses/algorithms-data-structures-part-one/transcript • https://app.pluralsight.com/library/courses/algorithms-data-structures-part-two/transcript	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	• https://edutechlearners.com/download/Introduction_to_algorithms-3rd%20Edition.pdf
C# Samples	• https://www.youtube.com/watch?v=J9dqgrOSR2M • https://www.youtube.com/watch?v=K0-qs--naUo • https://www.youtube.com/watch?v=iKDhgVoXVTk

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

	• Traverse/Print
5	Implement your own version of the N-child tree with mentioned functionalities: • Insert • Delete • Contains • Get Elements by value • Get Elements by level • Iterator Breadth First • Iterator Depth First • Traverse/Print Breadth First • Traverse/Print Depth First
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	• Use of similar data structures already present in the language/framework is not allowed. • Make use of object-oriented capabilities for implementing the data structures. • Exception handling is expected in the program
Evaluation Criteria	• Code Completeness and Correctness • Usage of OO Principles, package/class structure, class/function/variable names • Complexity of the operations • Code should be in running condition • Presentation Skills