Code Base Documentation

Mutation Testing of Source Code
DataStructures in JavaScript

Pranjal Walia, IMT2019062
Samaksh Dhingra, IMT2019075

Overview

The project is a library that exposes APIs for interacting with implementations of common data structures in JavaScript as it does not have a native template library.

The List of supported data structures is as follows:

1. Queue
2. Stack
3. Singly Linked List
4. Binary Search Tree (BST)

Prerequisites

The codebase is built on top of the Node.js runtime and requires a node version >=16 and the *yarn* package manager.

https://nodejs.org/en/about/
https://yarnpkg.com/

1. After entering the code folder, execute `yarn install` to install all the dependencies of the folder
2. Unit tests can be run by executing `yarn test` in the code folder
3. Mutation tests can be run by executing `yarn test:mutation` in the code folder
4. Unit test coverage reports can be generated by running `yarn coverage`

Directory Structure

After unzipping the submission file, the folder titled ***code*** contains all the relevant source code accompanied with the corresponding unit tests for the same, following directories lie within code:

1. The folder titled as ***src*** houses all the implementations, and the name of each file corresponds to the data structure it implements.
2. The folder titled as ***test*** houses all tests for the corresponding implementations and the name of each file corresponds to the tests for that data structure.
3. The folder titled as ***coverage/lcov-report*** houses the code coverage report of the unit tests within the file *index.html*
4. The folder titles as ***reports/mutation*** houses a file *mutation.html* that contains the source code mutation analysis report.

Other relevant files include the following:

1. Gruntfile.js : Grunt is a task runner for the Node.js ecosystem and is used to orchestrate the testing and coverage generation for all the different sub-modules, this file contains the configuration for the same.
   https://gruntjs.com/
2. stryker.conf.json : Stryker is the library used for facilitating the automatic mutation of source code. Said file contains the relevant configuration of that tool.
   https://stryker-mutator.io/

Sub-Modules and functions

1. LinkedList: *src/linkedList.js, test/linkedList.test.js*
   a. head()
   b. count()
   c. insertFirst()
   d. insertLast()
   e. insertAt()
   f. removeFirst()
   g. removeLast()
   h. removeAt()
   i. getByIndex()
   j. search()
   k. reverse()
   l. findMid()
   m. mergeSort()
   n. isEmpty()
   o. clear()
   p. fromArray()
2. Queue: *src/queue.js, test/queue.test.js*
   a. enqueue()
   b. dequeue()
   c. front()
   d. back()
   e. pop()
   f. fromArray()
   g. reverse()
   h. sort()
   i. reverseFirstK()
   j. size()
   k. clone()
   l. isEmpty()
   m. toArray()
   n. clear()
   o. interleave()

3. Stack: *src/stack.js, test/stack.test.js*
    a. push()
    b. peek()
    c. pop()
    d. sortStack()
    e. isEmpty()
    f. size()
    g. toArray()
    h. fromArray()
    i. slidingMaxOfKSubarrays()
    j. printNGE()
    k. computeHistogram()
4. BST*: src/binarySearchTree.js, test/binarySearchTree.test.js*
    a. constructPreOrder()
    b. insertKey()
    c. containsKey()
    d. treeHeight()
    e. max()
    f. min()
    g. findKey()
    h. sum()
    i. root()
    j. removeKey()
    k. inOrder()
    l. preOrder()
    m. postOrder()