**Documentation – WordBoard.java**

1. **Task**

Given a 2D board and a word, find if the word exists in the grid. The word can be constructed from letters of sequentially adjacent cell, where “adjacent” cells are those horizontally or vertically neighbouring.

*Note: The same letter cell may not be used more than once.*

1. **Entry point and the Inputs**

**Inputs:**

* A 2D word board

**char [][] wordboard = new char [][]**

**{**

**{'A','B','C','E'},**

**{'S','F','C','S'},**

**{'A','D','E','E'},**

**};**

* A word to find

**String word = "ABFC";**

**Entry point of the program:**

public static void main(String[] args)

**Output**: true (*for words that exist*) or false (*for word that do not exist*) – see the task description

1. **Method descriptions:**

|  |  |
| --- | --- |
| **Method** | **Description** |
| public static boolean find\_word (…) | *The primary method that checks the words existence/non-existence.*  Inputs:  word board, visited (type: 2d array) , word (type: 1d array)  Output:  Returns true/false (type: boolean) |
| public static boolean word\_run (…) | *The method that runs through the word board and finds existence/non-existence of each letter of the word.*  Inputs:  word board, visted (type: 2d array), word (type: 1d array), idx, i, j (type: int)  Output:  Returns true/false (type: boolean) |
| public static boolean valid (…) | Validate the run of the characters.  Inputs:  word board (type: 2d array), i, j (type: int)  Output:  Returns true/false (type: boolean) |