

SOFTWARE ENGINEERING HW2 – gr485

Description of the test cases.

- The code is written in a2.js
- The test cases are written in codespec.spec.js
- The input file is trained.csv which is also present in the zip folder
- So, the path for the input file is given in the code, based on the file we display the valid cell ranges for the CSV and also the operations that the user can give in the formula.

I have written **35** test cases, handling the possible errors and result verification at each stage.

Readcsvfile

1. should throw an error for an invalid file format when reading CSV
2. should throw an error if the given file does not exist
3. should throw an error if the given file is empty

Validate formula

4. should throw an error if the formula does not start with "="
5. should throw an error if the formula does not match the expected pattern
6. should throw an error for an unsupported operation

retrieveValuesInRange

7. should log values for a valid range

isCellRangeValid

8. should return true for a valid cell range
9. should return false for an invalid cell range

computeSumInRange

10. should compute the sum of valid numeric values
11. should handle empty values array
12. should throw an error for a negative sum

computeAverageInRange

13. should compute the average of valid numeric values

- 14. should handle empty values array
- 15. should handle values array with no valid values
- 16. should throw an error when the average is zero
- 17. should throw an error when the average is negative

computeMaxInRange

- 18. should return the maximum value in the range
- 19. should handle negative values in the range
- 20. should handle a range with no valid values

computeMinInRange

- 21. should return the minimum value in the range
- 22. should handle negative values in the range
- 23. should handle a range with no valid values

computeTotalZeros

- 24. should return the count of zeros in the range
- 25. should handle zero values represented as strings
- 26. should handle zero values represented as numbers
- 27. should handle a range with no zero values
- 28. should handle a range with no valid values

get cell values

- 29. should throw an error for a non-numeric value
- 30. should throw an error for a negative value
- 31. should return values for a valid range

retrieveValuesInRange

- 32. should log values for a valid range

computeMultiplicationInRange

- 33. should compute the multiplication of valid numeric values
- 34. should throw an error when the product is negative
- 35. should handle empty values array
- 36. should throw an error when the product is zero

```
gayathri@Gayathris-Air SE_2-gr485 % npx jasmine /Users/gayathri/Desktop/SE_2-gr485/codespec.spec.js

Randomized with seed 71091
Started
.....testtttt
.....testtttt
.....

35 specs, 0 failures
Finished in 0.008 seconds
Randomized with seed 71091 (jasmine --random=true --seed=71091)
gayathri@Gayathris-Air SE_2-gr485 %
```

Above is the screenshot after all the testcases are executed.