

Assignment 5 – Testing the Word Term Frequencies Calculating Program

RunLog:

- Environment: Windows 10 OS | Run on March 1st 2018 | 11:00PM || Python pyc file was decompiled and ran as py file
- **As of March 2nd 2018**
 - BlackBoxTesting
 - Total Bugs found : **5 bugs (Test ID: 2, 5, 6, 9, 11)**
 - Total Issues found : **1 issue (Test ID: 3)**
 - Stress Testing
 - Total Issues found : **2 Issues (Test ID:)**
 - Combinational Testing
 - Internationalization Testing
- Please refer the table on the next page. **Bugs are red coded**

Test ID Execution Order Type	Summary	Input	Expected Output	Procedure	Test Type	Actual Output
1 correct	Summary information using input file	mike.txt containing "mike was here"	3 tokens 3 unique tokens	1. Execute the command "python tf.pyc -s -i mike.txt" 2. Validate output is correct.	Functional Validation	
2 Incorrect - Bug	Verbose Mode: -Used other language, say Tamil language in input file	input.json file containing "தமிழ்"	"Can process only English language statements" since I assume this program does not support different languages	1. Execute the command "python termFrequency.py -v" then "input.json" 2. Validate output is having info stating "Non-English language not supported"	Functional and Internationalization Validation	Vivek-Pc@kev MINGW64 /e/Marist/Semester2/SoftwareVerificationAndMaintenance/termFrequency_2_ \$ python termFrequency.py -v Please enter a file name: input.json Debug mode! Input file: input.json Whole Input File Content: தமிழ் ===== Output file: ./termFreqOutput.txt Term Frequencies: "தமிழ்": {"count": "1", "tokenIds": ["0"]} Traceback (most recent call last): File "termFrequency.py", line 170, in <module> output(i, wordKey[0]) File "termFrequency.py", line 41, in output outputFile.write("%s": {"count": "%s", "tokenIds": %s}' % (wordKey, len(sentenceDict[wordKey])), str(sentenceDict[wordKey]).replace("'", "")) File "C:\DEV\PYTHON3\lib\encodings\cp1252.py", line 19, in encode

						<pre> return codecs.charmap_encode(input,self.errors,encoding_table)[0] UnicodeEncodeError: 'charmap' codec can't encode characters in position 1-2: character maps to <undefined> </pre>
3 Feature Request Issue	Verbose Mode: -Used empty file	Input.json file contains nothing	Information stating like “No input found in the input.json file”	1. Execute the command “python termFrequency.py -v” then “input.json” 2. Validate output is having info stating “File contains no input”	Functional Testing	Vivek-Pc@kev MINGW64 /e/Marist/Semester2/SoftwareVerificationAndMaintenan ce/termFrequency_2_ \$ python termFrequency.py -v Please enter a file name: input.json Debug mode! Input file: input.json Whole Input File Content: ===== Output file: ./termFreqOutput.txt Term Frequencies:
4 Correct Output	Verbose Mode: -Used Code inside the file	Input.json file contains the python programing code	X tokens with X unique tokens	1. Execute the command “python termFrequency.py -v” then “input.json” 2. Validate that output is correct	Functional Testing	Used factorial python program inside the file and the output came with no issue. This is because, the pgm converts all the inputs inside the input.json file as string and then process them ignoring the symbols
5 Incorrect output Bug	Verbose Mode: -Used combination of symbols to verify the consistency of ignorance. The “_” is not being parsed well - The “_” should also be parsed like other symbols.	Input.json file contains “hello_hello hello hello hai-hai hai hai abc***abc abc abc pop(pop) pop pop”	X tokens with X unique tokens should be obtained	1. Execute the command “python termFrequency.py -v” then “input.json” 2. Validate that the output	Functional Testing	Vivek-Pc@kev MINGW64 /e/Marist/Semester2/SoftwareVerificationAndMaintenan ce/termFrequency_2_ \$ python termFrequency.py -v Please enter a file name: input.json Debug mode! Input file: input.json Whole Input File Content: hello_hello hello hello hai-hai hai hai

				2. Verify that the given output file is obtained with no issues		Time Before completing the file write function: 2018-03-02 12:41:38.581536 Time After completing the file write function: 2018-03-02 12:41:38.586522
9 Incorrect Output Bug	<p>Output file mode usage with other options -d -s</p> <p>-Used the output file mode along with other options like -d or -s or both -d -s along with -o while executing the command, the given output file not created</p> <p>- For example,</p> <p>Positive case: refer 8 above</p> <p>\$ python termFrequency.py -o output.txt</p> <p>-When above command used, the output.txt file is created with no issues.</p> <p>Negative case:</p> <p>\$ python termFrequency.py -o -d -s output.txt</p> <p>-Now, when this command executed, the output.txt file is not created.</p>	Input.txt contains "Hello hai hello world hai hello"	X tokens	1. Execute the command "python termFrequency.py -v" then "input.json" and validate the output	Functional Testing	<p>Vivek-Pc@kev MINGW64</p> <p>/e/Marist/Semester2/SoftwareVerificationAndMaintenance/termFrequency_2_</p> <p>\$ python termFrequency.py -o -d -s output.txt</p> <p>Time Before inputting the fileName: 2018-03-02 12:52:58.017764</p> <p>Please enter a file name: input.txt</p> <p>Time Before completing the file write function: 2018-03-02 12:53:00.341114</p> <p>Time After completing the file write function: 2018-03-02 12:53:00.346119</p> <p>Summary:</p> <p>Tokens: 9</p> <p>Unique Tokens: 6</p> <p>Max Fequency: ('hello', 3)</p> <p>Min Fequency: 1</p>
10 Correct Code	<p>Verbose mode with -a sorting feature testing</p> <p>-Used the input.txt file containing the unsorted words</p>	Input.txt contains "	X tokens	1. Execute the command "python termFrequency.py -v" then "input.json" and validate the output	Functional Tesging	<p>Vivek-Pc@kev MINGW64</p> <p>/e/Marist/Semester2/SoftwareVerificationAndMaintenance/termFrequency_2_</p> <p>\$ python termFrequency.py -v</p> <p>Time Before inputting the fileName: 2018-03-02 12:57:01.796715</p> <p>Please enter a file name: input.txt</p> <p>Debug mode!</p> <p>Input file: input.txt</p> <p>Whole Input File Content: Hello Ada Joey Ada Hello</p> <p>=====</p> <p>Output file: ./termFreqOutput.txt</p>

						<p>Term Frequencies: "hello": {"count": "2", "tokenIds": ["0", "4"]} "ada": {"count": "2", "tokenIds": ["1", "3"]} "joey": {"count": "1", "tokenIds": ["2"]} Time Before completing the file write function: 2018-03-02 12:57:03.437732 Time After completing the file write function: 2018-03-02 12:57:03.442211</p> <p>Vivek-Pc@kev MINGW64 /e/Marist/Semester2/SoftwareVerificationAndMaintenance/termFrequency_2_ \$ python termFrequency.py -v -a Time Before inputting the fileName: 2018-03-02 12:57:32.105676 Please enter a file name: input.txt Debug mode! Input file: input.txt</p> <p>Whole Input File Content: Hello Ada Joey Ada Hello</p> <p>=====</p> <p>Output file: ./termFreqOutput.txt</p> <p>Term Frequencies: "ada": {"count": "2", "tokenIds": ["1", "3"]} "hello": {"count": "2", "tokenIds": ["0", "4"]} "joey": {"count": "1", "tokenIds": ["2"]} Time Before completing the file write function: 2018-03-02 12:57:36.949347 Time After completing the file write function: 2018-03-02 12:57:36.954351</p>
11 Incorrect Output Bug	Input Mode -Used the input mode and inputted the words	Inputting in the command console like	X tokens	1. Execute the command "python termFrequency.py -v" then	Functional Testing	Vivek-Pc@kev MINGW64 /e/Marist/Semester2/SoftwareVerificationAndMaintenance/termFrequency_2_

	<ul style="list-style-type: none"> - The input is being received again and again - There should be a way to stop accepting and process the input given. 	"Hello world hello world still accepting the input not valid...!!!"		"input.json" and validate the output		<pre>\$ python termFrequency.py -n Hello world hello world still accepting the input not valid...!!! Traceback (most recent call last): File "termFrequency.py", line 125, in <module> content = sys.stdin.read() KeyboardInterrupt</pre>

STRESS TESTING

Test ID	Execution Order Type	Summary of what I did	Input	Expected Output	Procedure	Actual Output Obtained: (trimmed output a little to be view legibly)
1	1 Correct Output	Normal file with normal inputs	Input.txt contains one line	X tokens	1. Execute the command "python termFrequency.py -v" then "input.json" and validate the output	Vivek-Pc@kev MINGW64 /e/Marist/Semester2/SoftwareVerificationAndMaintenance/termFrequency_2_ \$ python termFrequency.py -v Please enter a file name: input.txt Debug mode! Input file: input.txt Whole Input File Content: "hello world hai hello wolod world hello hello" =====

						<p>Output file: ./termFreqOutput.txt</p> <p>Term Frequencies: "hello": {"count": "4", "tokenIds": ["0", "3", "6", "7"]} "world": {"count": "2", "tokenIds": ["1", "5"]} "hai": {"count": "1", "tokenIds": ["2"]} "wolod": {"count": "1", "tokenIds": ["4"]}</p> <p>Time Before completing the file write function: 2018-03-02 12:06:06.435161 Time After completing the file write function: 2018-03-02 12:06:06.436153</p>
2	2 Correct Output	Small file testing	Input.txt contains 122 words	X tokens	<p>1. Execute the command "python termFrequency.py -v" then "input.json" and validate the output</p>	<p>Vivek-Pc@kev MINGW64 /e/Marist/Semester2/SoftwareVerificationAndMaintenance/termFrequency_2_ \$ python termFrequency.py -v Please enter a file name: input.txt Debug mode! Input file: input.txt</p> <p>Whole Input File Content: Stress testing is a type of performance testing focused on determining an application's robustness, availability, and reliability under extreme conditions. The goal of stress testing is to identify application issues that arise or become apparent only under extreme conditions. These conditions can include heavy loads, high concurrency, or limited computational resources. Proper stress testing is useful in finding synchronization and timing bugs, interlock problems, priority problems, and resource loss bugs. The idea is to stress a system to the breaking point in order to find bugs that will make that break potentially harmful. The system is not expected to process the overload without adequate resources, but to behave (e.g., fail) in an acceptable manner (e.g., not corrupting or losing data).</p>

						<p>=====</p> <p>Output file: ./termFreqOutput.txt</p> <p>.....</p> <p>Time Before completing the file write function: 2018-03-02 12:18:31.055440</p> <p>Time After completing the file write function: 2018-03-02 12:18:31.067437</p>
3	3 Correct Output	Big file testing	Input.txt contains 122,000 words	X tokens	1. Execute the command "python termFrequency.py -v" then "input.json" and validate the output	<p>Time Before completing the file write function: 2018-03-02 12:21:47.030644</p> <p>Time After completing the file write function: 2018-03-02 12:21:47.035623</p>
4	4 Correct Output	Bigger file testing	Input.txt contains 3,880,800 words with 50,409 lines	X tokens	1. Execute the command "python termFrequency.py -v" then "input.json" and validate the output	<p>Time Before inputting the fileName: 2018-03-02 12:38:41.372525</p> <p>Time Before completing the file write function: 2018-03-02 12:39:23.800657</p> <p>Time After completing the file write function: 2018-03-02 12:39:23.805682</p>
5	5 Memory Error – I guess Issue is with my system	Bigger file testing	Input.txt contains 17,791,500 words with 147,918 lines	X tokens	1. Execute the command "python termFrequency.py -v" then "input.json" and validate the output	<p>=====</p> <p>Traceback (most recent call last):</p> <p>File "termFrequency.py", line 140, in <module></p> <p>tokens = re.findall('[\\w]+' , content.lower().replace('\\n', ' ').replace('\\r', ' ').replace('.', ' ').replace(';', ' ').replace(',', ' '))</p> <p>MemoryError</p>
6	6 Memory Error – I guess Issue is	Bigger file testing	Input.txt contains 24,766,000 words with	X tokens	1. Execute the command "python termFrequency.py -v" then "input.json" and validate the output	<p>Traceback (most recent call last):</p> <p>File "termFrequency.py", line 140, in <module></p> <p>tokens = re.findall('[\\w]+' , content.lower().replace('\\n', ' ').replace('\\r', ' ').replace('.', ' ').replace(';', ' ').replace(',', ' '))</p>

	with my system		203,204 lines			MemoryError