Introduction to Software Engineering(ISAD1000) - Final Assessment

Name : Thilina Prasad Athukorala

ID : 21038875

Content

- Introduction
- Module descriptions
- Modularity
- Black-box test cases
- White-box test cases
- Test implementation and execution
- Version control
- Ethics
- Discussion

Introduction

This programme is a system of String conversions which include Converting a given string to upper case or lower case, Identify whether numeric values are in a given string, Identify whether a given string is a valid number or not, Remove any numeric values in a given string and then convert the string to upper case or lower Case and conversions of mass measurements which includes Converting a number which represents a mass given in kilograms to pounds and vice versa, and mass given in milligram to ounce and vice versa.

Module Descriptions

Main method

• Describes the menu displayed for the user to select.

• Calls other methods in the system by considering the user choice from the menu.

stringCase method

• This method converts a string value to uppercase or lower case.

•Perform necessary functions according to the parameter input.

numScan method

• Scans the string to check whether it contains any digits.

• Assumed that only one string is entered (Not a set of string separated in spaces).

• Assumed that decimal numbers are not entered.

• Designed the method to obtain input string from the user as the data input.

• Split the string input into array of characters and scan each of them separately.

• If the character is a digit, then a count is taken to detect whether the count is zero at the end to determine if the string contains digits.

• Negative numbers are detected by '-' sign.

• Even if '+' is entered it is detected as a positive number.

valNum method

• String input for the method is given using a file.

• Split the input String into an array of characters and scan them to detect whether the entered String is a valid numerical number.

• Just like in the previous method, take a digit count to check whether the number of characters in the string is equal to the digit count at the end.

• If the above conditions are fulfilled it confirms that the string is a number.

• If a decimal number is entered it is detected by the '.' Character.

• It is detected as a decimal number only if the '.' Character count is one and the total digit count is equal to one less than the number of characters in the string (since one character is allocated to '.' symbol).

remDigit method

• User inputs a string value.

• The string is split into an array of characters and checked for digits in it.

• If any digits are found, they are replaced with an empty Character.

• Then the rest of the characters are collected into a variable to form the new string without any digits since they were replaced with an empty character.

• Then the digit-less string is converted to uppercase and lower case.

massConversion method

• User selects which conversion to be done from the 4 given selections.

• Selections are

1. Convert values in Kilograms to pounds.
2. Convert values in pounds to Kilograms.
3. Convert values in milligrams to ounces.
4. Convert values in ounces to milligrams.

• Next the values to be converted are present in a file named 'calcData.txt'.

• The string values are extracted and converted to decimal values.

• The decimal values are then converted to the relevant mass unit and displayed to the user written to another file named 'calculated.txt'.

• All the converted values are collected and printed to the file at the end.

• filePrint method is used to print the values to the new file after collecting it to the variable 'fileData'.

• This method was checked in a separate file as the input is entered to a newly created file.

• Before the file writing method is implemented the method was created to get a user input and display results to the screen.

Modularity

a.) Code is compiled and it runs without any syntax errors.

b.) Checklist

Checklist

1. Are global variables being used?

2. Does the code contain sequential tasks?

3. Does the code contain only an appropriate number of parameters (Not more than 6)?

4. Does the code contain flag controls?

5. Does the code contain any merged methods which are not related?

6. Does the code contain excess return values?

7. Does the code contain any reused parts which are effective in the programme?

8. Are there any methods which are subsets which are included in other methods?

9. Is the code simple so that programme can be understood by a person with moderate level of knowledge?

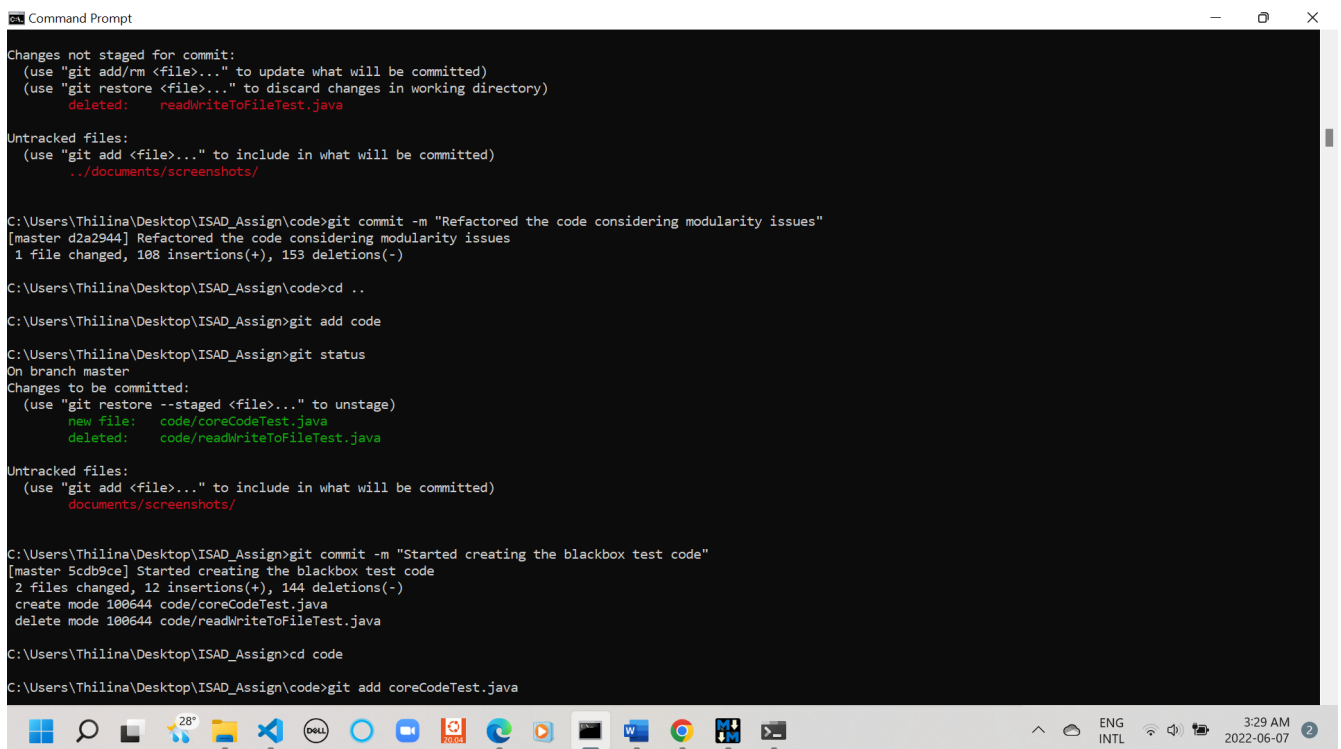10. Does the code consist of tight coupling between modules?

c.) Code review

Code review

1. Initially user inputs and user selections were all given using a global variable, the scanner method was implemented once for the whole class. Code was refactored to use separate variables in each method and initialized in it.

2. Code does not contain any sequential tasks.

3. Code does not contain excess number of parameters.

4. Code does not contain any control flags.

5. Code does not contain any merged methods which are not related.

6. Code contains return values to implement testcases.

7. Yes, Code contains few duplicated pieces of codes which are used effectively.

8. Converting the string to uppercase is a method merged into the method which removes digits from a string even though the code piece is repeated.

9. Only complex piece of code in it is the file print method since all the other methods are written of basic concepts and language.

10. Coupling has been reduced to the minimum by separating almost all the methods performing all their tasks individually without any merging process.

d.)**Refactoring issues**

• All global variables were removed and Separate variables for each method is implemented.

• Even though case conversion method was merged into the case conversion after digit removal; it was separated.

Blackbox testing

Category	Test Data	Expected Result
selection =1	1, 1, thilina	Thilina
selection =2 .	2, thi210li388na .	"The entered string contain numerical values"

Category	Test Data	Expected Result
2, athukorala . .	"The entered string does not contain any numerical values"	
selection =3 . . .	3, 210388	"It is a valid number"
3,388.75	"It is a decimal number"	
3,38875thilina	"It is not a valid number"	
selection =4	4, 1, Athukorala21038875	"value Converted to uppercase : ATHUKORALA value Converted to Lowercase : athukorala"
selection = 6	6 .	"Error! Invalid Selection in main menu Available choices : 1/2/3/4/5"

Test implementation and execution

Module name	BB test design (EP)	BB test design (BVA)	WB test design	EP test code (implemented/ run)	BVA test code (implemented /run)	White-Box testing (implemented/run)
stringCase	Done	Not done	Not done	done	Not done	Not done

Module name	BB test design (EP)	BB test design (BVA)	WB test design	EP test code (implement ed/ run)	BVA test code (implemented /run)	White-Box testing (implemented/run)
numScan	Done	Not done	Not done	Done	Not done	Not done
valNum	Done	Not done	Not done	Done	Not done	Not done
remDigit	Done	Not done	Not done	Done	Not done	Not done
massConversion	Not done	Not done	Not done	Not done	Not done	Not done

Git log

0000000000000000000000000000000000000000 6f7eae467c71e6007f3fd4699b103c8dccde05d0 Thilina thilinaprasad603@gmail.com 1654268697 +0530 commit (initial): Created the 2 folders and sub directory 6f7eae467c71e6007f3fd4699b103c8dccde05d0 377ccc8b3162dc923b3e0f7469034161e620b2d7 Thilina thilinaprasad603@gmail.com 1654306638 +0530 commit: Introduction to the code 377ccc8b3162dc923b3e0f7469034161e620b2d7 4d381c32a998721e0d08daeb9bcd56b389bb4337 Thilina thilinaprasad603@gmail.com 1654306843 +0530 commit: imported packages and introduced main method 4d381c32a998721e0d08daeb9bcd56b389bb4337 210db5c506a5ceb1db8240f0d29ec603f3450519 Thilina thilinaprasad603@gmail.com 1654307272 +0530 commit: created the list of methods to be handled specifying what each method does 210db5c506a5ceb1db8240f0d29ec603f3450519 ca6315e8fe499ceee30a1162d9f0554162016d03 Thilina thilinaprasad603@gmail.com 1654307581 +0530 commit: initialized scanner and variables to obtain user input ca6315e8fe499ceee30a1162d9f0554162016d03 6e0e183426c3a95916a37678658e4f1d77b576d5 Thilina thilinaprasad603@gmail.com 1654307769 +0530 commit: created the menu to obtain user choice for case conversion of string and obtain scanner input 6e0e183426c3a95916a37678658e4f1d77b576d5 c0e18348e26584d4fdf8b6b5a794ea5e6e604cc8 Thilina thilinaprasad603@gmail.com 1654308398 +0530 commit: created first If condition to convert user input string to uppercase c0e18348e26584d4fdf8b6b5a794ea5e6e604cc8 fc6d1ffad90ee6714c232a27c7f36e529329d2f2 Thilina thilinaprasad603@gmail.com 1654308605 +0530 commit: created second If condition to convert user input string to Lowercase fc6d1ffad90ee6714c232a27c7f36e529329d2f2 cefa6818b166fc4e33306415439d70309f04a89e Thilina thilinaprasad603@gmail.com 1654308774 +0530 commit: completed first method which converts user input to lower or upper case and handled the condition if user inputs an invalid selection cefa6818b166fc4e33306415439d70309f04a89e 5e2e047e8d3e1d60ea52515acb49746dfdd1938b Thilina thilinaprasad603@gmail.com 1654309107 +0530 commit: created a branch to test the second function 5e2e047e8d3e1d60ea52515acb49746dfdd1938b cefa6818b166fc4e33306415439d70309f04a89e Thilina thilinaprasad603@gmail.com 1654309129 +0530 checkout: moving from master to method2 cefa6818b166fc4e33306415439d70309f04a89e 9d1b5ae3e69fef18d26fb835fa87df3b805aea14 Thilina thilinaprasad603@gmail.com 1654309300 +0530 commit: removed global scanner in branch 9d1b5ae3e69fef18d26fb835fa87df3b805aea14 b2ec1b5928b34af0c7ee7e72f38afddd9254965f Thilina thilinaprasad603@gmail.com 1654309461 +0530 commit: declared variables required for 2nd method and initialized b2ec1b5928b34af0c7ee7e72f38afddd9254965f b26b54acddfb689107b5505c5ed687719528e9fe Thilina thilinaprasad603@gmail.com 1654309690 +0530 commit: used a parameter input instead of user input to test wether the character array is working b26b54acddfb689107b5505c5ed687719528e9fe

f90bb301b97fbe11cc659f277e0fbb676411ed2b Thilina thilinaprasad603@gmail.com 1654309810 +0530 commit: checking wether the character array works after completing the method f90bb301b97fbe11cc659f277e0fbb676411ed2b 8856fa39a4b9e2c530bf9b0b076a408331a87fac Thilina thilinaprasad603@gmail.com 1654309901 +0530 commit: removed the other methods in main method to check the array 8856fa39a4b9e2c530bf9b0b076a408331a87fac c1f0b98d5874fb10f0e233c9c6b7d5dd409005ec Thilina thilinaprasad603@gmail.com 1654310060 +0530 commit: removed string case method in branch to check the array assigning it as main method c1f0b98d5874fb10f0e233c9c6b7d5dd409005ec 1dccea839215cb302015bcb4d0917687d810ad81 Thilina thilinaprasad603@gmail.com 1654310153 +0530 commit: Array check successfull 1dccea839215cb302015bcb4d0917687d810ad81 5e2e047e8d3e1d60ea52515acb49746dfdd1938b Thilina thilinaprasad603@gmail.com 1654310243 +0530 checkout: moving from method2 to master 5e2e047e8d3e1d60ea52515acb49746dfdd1938b 1dccea839215cb302015bcb4d0917687d810ad81 Thilina thilinaprasad603@gmail.com 1654310290 +0530 checkout: moving from master to method2 1dccea839215cb302015bcb4d0917687d810ad81 48785900e5d49f360263ae1ba1aabfd834de6da8 Thilina thilinaprasad603@gmail.com 1654310413 +0530 commit: added back the main method and string case method 48785900e5d49f360263ae1ba1aabfd834de6da8 39d29a463f48e9bf1e218aa96f8ea8d857b25352 Thilina thilinaprasad603@gmail.com 1654310573 +0530 commit: Edit the branch to recreate the array focusing one String instead of a sentence since other methods focus on a single string 39d29a463f48e9bf1e218aa96f8ea8d857b25352 607c7764056b4e46af715d03adb37ee093ce79d1 Thilina thilinaprasad603@gmail.com 1654310780 +0530 commit: change the array to identify if negative or positive signs are entered by user 607c7764056b4e46af715d03adb37ee093ce79d1 d28f050c48d768cf2256ebb00ac7f1562a1b638e Thilina thilinaprasad603@gmail.com 1654310971 +0530 commit: successfullt finished numscan method in branch,ready to merge d28f050c48d768cf2256ebb00ac7f1562a1b638e 5e2e047e8d3e1d60ea52515acb49746dfdd1938b Thilina thilinaprasad603@gmail.com 1654310987 +0530 checkout: moving from method2 to master 5e2e047e8d3e1d60ea52515acb49746dfdd1938b 97f05a7bd6d081cd5fd49578520ebcef8531a457 Thilina thilinaprasad603@gmail.com 1654311415 +0530 commit (merge): merge conflict resolved 97f05a7bd6d081cd5fd49578520ebcef8531a457 7824a34ae5bcd7126f176ff74084102b8fdfd293 Thilina thilinaprasad603@gmail.com 1654311629 +0530 commit: resolved unwanted merge changes 7824a34ae5bcd7126f176ff74084102b8fdfd293 7824a34ae5bcd7126f176ff74084102b8fdfd293 Thilina thilinaprasad603@gmail.com 1654312221 +0530 checkout: moving from master to Fileread 7824a34ae5bcd7126f176ff74084102b8fdfd293 53ac355ea251d365f1b4a1897a8aff6ea232e3e0 Thilina thilinaprasad603@gmail.com 1654312433 +0530 commit: initiated a method to test whether the file is read properly 53ac355ea251d365f1b4a1897a8aff6ea232e3e0 989d9640af9f2f3e3befeaac5cdccaac5da6aa3e Thilina thilinaprasad603@gmail.com 1654312910 +0530 commit: Successfully ectracted data from file 989d9640af9f2f3e3befeaac5cdccaac5da6aa3e f52b759fb0337ee82dde5803276175b3c2ed271c Thilina thilinaprasad603@gmail.com 1654313105 +0530 commit: Started modifying for the file reading method to suit the required function f52b759fb0337ee82dde5803276175b3c2ed271c da7edffc7da602ec84a8e88e4d69b21b395eeead Thilina thilinaprasad603@gmail.com 1654313384 +0530 commit: developed the method to identify integers and decimals by character scan da7edffc7da602ec84a8e88e4d69b21b395eeead 1b05d27267b263d795ac10072a1ac739bcc7ae27 Thilina thilinaprasad603@gmail.com 1654313660 +0530 commit: Included exception handling 1b05d27267b263d795ac10072a1ac739bcc7ae27 a12b0a2743a25e7b220f6b317d4981152f27c665 Thilina thilinaprasad603@gmail.com 1654313808 +0530 commit: Branch developed successfully for 3rd method in system a12b0a2743a25e7b220f6b317d4981152f27c665 7824a34ae5bcd7126f176ff74084102b8fdfd293 Thilina thilinaprasad603@gmail.com 1654313817 +0530 checkout: moving from Fileread to master 7824a34ae5bcd7126f176ff74084102b8fdfd293 a12b0a2743a25e7b220f6b317d4981152f27c665 Thilina thilinaprasad603@gmail.com 1654313988 +0530 merge Fileread: Fast-forward a12b0a2743a25e7b220f6b317d4981152f27c665 1653ea353253e133bc2e54993fdee81f73132460 Thilina thilinaprasad603@gmail.com 1654314138 +0530 commit: Successfully merged the Filerread branch which contain the 3rd method 1653ea353253e133bc2e54993fdee81f73132460 641a8469be1b52cc9461a58a7f1f9d5eb3ad4028 Thilina thilinaprasad603@gmail.com 1654314456 +0530

commit: Initiated the 4th method creation to remove digits in a string 641a8469be1b52cc9461a58a7f1f9d5eb3ad4028 c9da6dd1dd2afb476c20022fff7cceab736cbc98 Thilina thilinaprasad603@gmail.com 1654314642 +0530 commit: created variables required for the 4th method c9da6dd1dd2afb476c20022fff7cceab736cbc98 c0dfc4cbd9557d529357d99fb3f5921289f3c3cd Thilina thilinaprasad603@gmail.com 1654314802 +0530 commit: created an array to detect digits and replace them with an empty character c0dfc4cbd9557d529357d99fb3f5921289f3c3cd 51c854200403d8c22bf866493159a5a1a33b51b8 Thilina thilinaprasad603@gmail.com 1654315053 +0530 commit: Successfully completed method 4 by adding a similarmethod as stringCase after removing digits in the string 51c854200403d8c22bf866493159a5a1a33b51b8 6db81ec0ff773a4325ce433cc5deb1ccce10372f Thilina thilinaprasad603@gmail.com 1654315652 +0530 commit: created a test file to test file input and writing to a file for method 5 (mass calculations) 6db81ec0ff773a4325ce433cc5deb1ccce10372f 60023e097924a51971b65c609dab64b3491cb571 Thilina thilinaprasad603@gmail.com 1654315905 +0530 commit: modified the test file to create the method of writing to file 60023e097924a51971b65c609dab64b3491cb571 d241a2cc9cf4cc25020c73fa138f42d9d0c05fcb Thilina thilinaprasad603@gmail.com 1654316181 +0530 commit: added a seperate method to print to a file in the test file d241a2cc9cf4cc25020c73fa138f42d9d0c05fcb 460030dca88fae0e76190f5e16575b1d6c0f5dcb Thilina thilinaprasad603@gmail.com 1654316440 +0530 commit: changed the test file since the previous code replaces each record while reading 460030dca88fae0e76190f5e16575b1d6c0f5dcb 96625288a5cc38447dd6936c257c12e76b9ab634 Thilina thilinaprasad603@gmail.com 1654316563 +0530 commit: Implemented the 5th method using the test file 96625288a5cc38447dd6936c257c12e76b9ab634 744f574a5d33c5cf46c973ebbee9b5aa6041ce31 Thilina thilinaprasad603@gmail.com 1654316767 +0530 commit: convert the string values to decimal values to use them for calculations 744f574a5d33c5cf46c973ebbee9b5aa6041ce31 891a2b415873a309c6550584f934f03eb14b12ff Thilina thilinaprasad603@gmail.com 1654317145 +0530 commit: created the conditions inside the loop and values are rounded to limit the decimal places 891a2b415873a309c6550584f934f03eb14b12ff 1806f376e5b3dc2823682782052fe8f39aab101b Thilina thilinaprasad603@gmail.com 1654317330 +0530 commit: Successfully finished method5 1806f376e5b3dc2823682782052fe8f39aab101b 5315be6a38d60734a6e8dd27e716db0507a19dc0 Thilina thilinaprasad603@gmail.com 1654317560 +0530 commit: Completed the write to file method 5315be6a38d60734a6e8dd27e716db0507a19dc0 8745055bd30e653bd9d3c0fcabc95f80eb072f53 Thilina thilinaprasad603@gmail.com 1654317680 +0530 commit: create user interface to display initial menu 8745055bd30e653bd9d3c0fcabc95f80eb072f53 bac132834f0855cda7a4efb5a8a98517de934965 Thilina thilinaprasad603@gmail.com 1654318038 +0530 commit: finished linking user choice with method initialization bac132834f0855cda7a4efb5a8a98517de934965 57e6e1692befb81ab9f6a6cd0634670c1b00a51e Thilina thilinaprasad603@gmail.com 1654318417 +0530 commit: Corrected errors 57e6e1692befb81ab9f6a6cd0634670c1b00a51e 46ae92654733c20e86573c165f23f623fb46edab Thilina thilinaprasad603@gmail.com 1654318584 +0530 commit: compiled the core code successfully 46ae92654733c20e86573c165f23f623fb46edab f4363129d7f6fc61288d6fef0dce1594ea493ba8 Thilina thilinaprasad603@gmail.com 1654363249 +0530 commit: started creating the report f4363129d7f6fc61288d6fef0dce1594ea493ba8 2154c23fd60d11b3cff1b7f3245c28ba6933d823 Thilina thilinaprasad603@gmail.com 1654414075 +0530 commit: Refactored the code using the modularity principles,removing global variables and other techniques in review of the code 2154c23fd60d11b3cff1b7f3245c28ba6933d823 67ade262b6303759a1b6e61087c33ad23c6ca1ea Thilina thilinaprasad603@gmail.com 1654414208 +0530 commit: Compiled the code after refactoring using modularity principles to check for syntax errors 67ade262b6303759a1b6e61087c33ad23c6ca1ea 4a60fc585b28a5ab1b3f1c4a888da5a8b2c7c0db Thilina thilinaprasad603@gmail.com 1654542539 +0530 commit: created main method module description 4a60fc585b28a5ab1b3f1c4a888da5a8b2c7c0db 7c8b51df82c1738e22f914fd161aab60a6b458ad Thilina thilinaprasad603@gmail.com 1654542734 +0530 commit: created stringCase method module description 7c8b51df82c1738e22f914fd161aab60a6b458ad 5da5e8f4757966f49a9b8b17b5bd308bed2e6f2a Thilina thilinaprasad603@gmail.com 1654542871 +0530 commit: created numScan method module description 5da5e8f4757966f49a9b8b17b5bd308bed2e6f2a 58f1865f1891e30d1de5d3268dff8bb88a40feb8 Thilina thilinaprasad603@gmail.com 1654542965 +0530 commit: created valNum method module description

58f1865f1891e30d1de5d3268dff8bb88a40feb8 224ad67001c003fd8ff678f3773af7af98acc677 Thilina thilinaprasad603@gmail.com 1654543083 +0530 commit: created remDigit method module description 224ad67001c003fd8ff678f3773af7af98acc677 649f388813d11e26fe67c22ba39ac6cae2cc7ff1 Thilina thilinaprasad603@gmail.com 1654543153 +0530 commit: created massConversion method module description 649f388813d11e26fe67c22ba39ac6cae2cc7ff1 6cbf7b532105921ee4d00742c607920b56121ac5 Thilina thilinaprasad603@gmail.com 1654543310 +0530 commit: Started writing modularity section and finished creating the checklist 6cbf7b532105921ee4d00742c607920b56121ac5 1868736c2395be53401dc8e31cbbe53fbc41a87a Thilina thilinaprasad603@gmail.com 1654543444 +0530 commit: Created the code review using the checklist questions 1868736c2395be53401dc8e31cbbe53fbc41a87a 37ea805694e84c9985c427046b8210f3071a0a86 Thilina thilinaprasad603@gmail.com 1654543634 +0530 commit: Completed ethics section (a) 37ea805694e84c9985c427046b8210f3071a0a86 f7251f114c43c10262d40a79160849aeae511b45 Thilina thilinaprasad603@gmail.com 1654543815 +0530 commit: Completed ethics section (b) f7251f114c43c10262d40a79160849aeae511b45 07f5df57a2f65dd6fadc2bbe2a65650414a022c8 Thilina thilinaprasad603@gmail.com 1654543895 +0530 commit: Completed the Discussion section 07f5df57a2f65dd6fadc2bbe2a65650414a022c8 d2a2944b2df85357ef66c2338c41a78f2432b730 Thilina thilinaprasad603@gmail.com 1654548877 +0530 commit: Refactored the code considering modularity issues d2a2944b2df85357ef66c2338c41a78f2432b730 5cdb9ce76fd48984ef2a1ff016c855de3d2f790d Thilina thilinaprasad603@gmail.com 1654549037 +0530 commit: Started creating the blackbox test code 5cdb9ce76fd48984ef2a1ff016c855de3d2f790d 388ec1458e11fc9e1c07bc17e86889749c8e98d7 Thilina thilinaprasad603@gmail.com 1654549234 +0530 commit: initialized test cases 388ec1458e11fc9e1c07bc17e86889749c8e98d7 edf0e4173bbb8adf7aada22e2c7864c0987bc6c8 Thilina thilinaprasad603@gmail.com 1654549287 +0530 commit: finished creating first test case edf0e4173bbb8adf7aada22e2c7864c0987bc6c8 9a8411f421d0394a65dffc1432edb02ea76f412b Thilina thilinaprasad603@gmail.com 1654549356 +0530 commit: finished creating second test case 9a8411f421d0394a65dffc1432edb02ea76f412b c7cad5fab278086d831bf94d787768cddf258bfd Thilina thilinaprasad603@gmail.com 1654549389 +0530 commit: finished creating third test case c7cad5fab278086d831bf94d787768cddf258bfd 45bf210c4a8f8b4fbefc33f40f511bd4d0fcef77 Thilina thilinaprasad603@gmail.com 1654549432 +0530 commit: finished creating fourth test case 45bf210c4a8f8b4fbefc33f40f511bd4d0fcef77 b5b21a1c9e3019b4a99498eaa3e25c3d7e05f675 Thilina thilinaprasad603@gmail.com 1654549554 +0530 commit: compiled and run the test code successfully b5b21a1c9e3019b4a99498eaa3e25c3d7e05f675 5a6ea5b16452252359fd82dd832f38f71816c796 Thilina thilinaprasad603@gmail.com 1654550014 +0530 commit: modified the report to suit the changes implmented 5a6ea5b16452252359fd82dd832f38f71816c796 b2267ac5638061ccddd0d7a3849a02cf8f0d8eb4 Thilina thilinaprasad603@gmail.com 1654550408 +0530 commit: added the calcData file which is read in the final method(mass conversion) b2267ac5638061ccddd0d7a3849a02cf8f0d8eb4 8e836390d90759baf0ba3eb45352c76272061910 Thilina thilinaprasad603@gmail.com 1654550602 +0530 commit: added file calculated.txt by running the 5th method since it creates a new file and display results in it 8e836390d90759baf0ba3eb45352c76272061910 83517b0842493bdd88e8a46374b6cec16eee43b1 Thilina thilinaprasad603@gmail.com 1654551327 +0530 commit: added more test cases 83517b0842493bdd88e8a46374b6cec16eee43b1 c5ba7fa7501abc1c5b085795c1d9fcb85166073c Thilina thilinaprasad603@gmail.com 1654551690 +0530 commit: changes c5ba7fa7501abc1c5b085795c1d9fcb85166073c 3297b4cb88694201b04c9816f208c9da00e9a41a Thilina thilinaprasad603@gmail.com 1654551821 +0530 commit: changed test cases

Ethics

a.) It can be assumed that this code is used in an organization dealing with mass-based products. If we assume that this is used in the process of ordering certain materials based on a certain mass unit such as Kilogram but all the required amounts are mentioned in pounds/milligrams/ounce by different clients. Then the amount of material to be imported are decided based on the calculation accuracy of this software. In

this case ethics play a major role since after the software is delivered to the client, they will be facing huge losses due to unmatched amount of imports in which imported amount is not equal to the requirement at that stage. The testing section should be done wisely and ethical behavior is mandatory for this to be done properly. Testing should be done in calculation section to focus on factors such as rounding to the correct decimal place and the integer conversions to double and maintaining decimals in calculation process without letting them to be converted to integers during any of the calculations furthermore it can be assumed that the customer names should be entered in uppercase format in all of the records and the case conversion section would be used. Then the proper tests should be done in order to handle all the type of conditions such as typing symbols instead of a certain characters, typing a number in between erroneously (can be detected by the numScan method and removed using remDigit method in the code) and other issues which are highly unexpected since the client should not be neglected at any cost due to any issue in the software. Since this software would be interacting directly with the client information and orders, it would be highly unethical to deliver a software with bugs and higher failure rate since the developer might not know whether he/she is a social worker who improve people's lives by helping with social and interpersonal difficulties, promoting human rights and wellbeing. Issue created for such a client would create a huge impact for many lives which create a highly unethical environment. As an example, we could use a client who distributes a huge amount of fertilizers free of charge to poor farmers who cannot afford to buy them. If an order of this client is missed to a certain software error it would impact all of those farmers lives creating a huge conflict situation.

b.)

• Enhancing quality of lives which have an impact from this software should be considered as in the example mentioned in (a) which could avoid the unethical behavior of the workers since they focus on the wellbeing of people rather than creating a piece of software to meet client desires as mentioned in ACS Code of Professional Conduct.

• Common unethical behavior is collecting information from the clients after delivering the software and maintenance procedures and selling that information to another similar organization (information of clients and import companies). These occur due to lack of honesty which is also an important section in ACS Code of Professional Conduct.

Discussion

Initially all the basic functional requirements were considered and implemented creating individual methods for each task. All methods were created and the menu for user selections were created after checking functions of methods. Global variables were used initially and most of the inputs were console inputs since it is easy to check the functionality of the software. After checking the basic functionality of selections and methods, a deeper process to check whether the calculations and outputs were 100% accurate according to the input given. First method was regarding the conversion of a string to uppercase or lowercase. In this section the software does not detect any symbols or non-alphabetic characters in it. Even though it just converts the detected letters to upper to lower case it could have been better to improve it to have those abilities as well. In the second method, String is checked for any numerical values. Even though this is designed for a single string input we could improve it to detect a whole sentence of set of characters separated by spaces which increases the scope of user imports and reliability. Another section to be developed is to detect decimal, positive and negative values (Developed it initially and recreated due to inability of continuing it in other methods). Third method was to detect if a string is a valid number. Fourth method was to remove any numerical values present in a string and output the new string without the detected numbers followed with a task to convert it to upper or lower case. Fifth method was regarding mass unit conversions which contained calculations to be checked wisely.