# Part 1 Understand the Problem

• Outline how the main logic would flow from inputs, to processing, to output.

Firstly, the program will ask user to input the length of the stick. Then, the menu comes out. User can choose one option from four candidate options buy entering numbers.

* The first option is to validate the sick. It will output if the difference is within the acceptable tolerance.
* The second option is to convert the input length from meters to centimeters.
* The third option is to convert the input length from meters to millimeters.
* The last option is to generate the report. It will output the stick length in 5 decimal places.

• Outline how the verify method would flow from start to return statement including math

Define **difference** as the difference between the input from the user and the expected length. To handle all situations, we use absolute value of **difference**. Therefore, we have

For any sticks, there is an acceptable tolerance. Therefore, the length of stick should be

We define **EPSILON** as the acceptable tolerance. Therefore, if the **difference** is equal to or smaller than **EPSILON**, we say stick is within tolerance of **EPSILON**. If the **difference** is greater than **EPSILON**, we say stick is not within tolerance of **EPSILON**.

Translate above in Java algorithm:

verify()

num **difference** = abs(length - EXPECTED\_LENGTH)

if **difference** <= **EPSILON**

output "Meter stick is within tolerance of EPSILON "

else

output "Meter stick is not within tolerance of EPSILON "

I will use Case Structure to process the program. The cases are the inputs of the user.

# Part 2a Pseudocode

## UML Class diagrams

Diagram

Description automatically generated

Figure 1. UML class diagrams

## Pseudocode for the verify method of class MeterStick

/\* Verify if the difference is within the tolerance \*/

public void verify(double length)

/\* absolute value of the difference between input length and expected length \*/

num difference = Math.abs(length - EXPECTED\_LENGTH);

/\* if the difference is within the tolerance, format and output information \*/

if (difference <= EPSILON)

output "Meter stick is within tolerance of %.5f\n", EPSILON

/\* if the difference is not within the tolerance, format and output information \*/

else

output "Meter stick is not within tolerance of %.5f\n", EPSILON

## Pseudocode for the toString method of class MeterStick

/\*\* Return the formatted String report. Numbers are formatted to 5 decimal places. /

public String toString()

String report = "MeterStick length " + String.format("adjacent %.5f, ", length)

Return report

## Pseudocode for the main method

Start

//declarations

num userLength

int option

String optput

num validateMeterStick = 1

num showInCentimeters = 2

num showInMillimeters = 3

num showReport = 4

Scanner keyboard

//initiation

optput = "Please select from one of the following:\n"

+ validateMeterStick + " to validate meter stick\n"

+showInCentimeters + " to show meter stick length in centimeters\n"

+showInMillimeters + " to show meter stick length in millimeters\n"

+showReport + " to show meter stick report"

output "Meter stick checker program."

output "Enter measured length in meters: "

/\*

\* This loop is used to make sure the input is a positive double.

\* If the input is not correct, the program will ask

\* the user to input again.

\*/

while true

input userLength

if userLength is a positive number

break

else

output "Invalid input. Please enter the correct length:"

// Output the menu

System.out.println(output);

// Create a MeterStick object

MeterStick stick = new MeterStick()

// Set the length

stick.setLength(userLength);

/\*

\* This loop is used to regulate the user's choice.

\* If the input is not correct, the program will ask

\* the user to input again

\*/

while true

input optionStr

// This is Java regular expression for 1, 2, 3, and 4

String regexOption = "^[1-4]\*$"

// Validate the input of option

boolean flagOption = optionStr.matches(regexOption)

// If the input is valid, proceed with the switch structure

if flagOption == true

/\* convert the input from String to int \*/

option = Integer. parseInt(optionStr)

case option

validateMeterStick: stick.verify(userLength)

showInCentimeters: output "Centimeters: %.5f\n",

stick.toCentimeters()

showInMillimeters: output " Millimeters: %.5f\n",

stick. toMillimeters()

showReport: output stick.toString()

end case

output "Program by Yanzhang Wu"

break

// If the input is not valid, ask user to enter the option again

else

output "Invalid menu option selected.Please enter the

correct option again:"

End

# Part 2b Flowchart

Diagram

Description automatically generated

Figure . Flowchart of verity method

Diagram

Description automatically generated

Figure . Flowchart of toString method

Diagram

Description automatically generated

Figure . Flowchart of main method

# Part 3 Test Plans for Algorithms

## Testing verify method of class MeterStick

|  |  |  |  |
| --- | --- | --- | --- |
| Length | Expected return | Actual return | Description |
| 0.99999 | Meter stick is within tolerance of 0.00010 | Meter stick is within tolerance of 0.00010 | Hand trace of method logic indicates that the method return value matches expectations. Length is in between tolerance range. |
| 1 | Meter stick is within tolerance of 0.00010 | Meter stick is within tolerance of 0.00010 | Hand trace of method logic indicates that the method return value matches expectations. Length is in between tolerance range. |
| 1.00005 | Meter stick is within tolerance of 0.00010 | Meter stick is within tolerance of 0.00010 | Hand trace of method logic indicates that the method return value matches expectations. Length is in between tolerance range. |
| 1.0002 | Meter stick is not within tolerance of 0.00010 | Meter stick is not within tolerance of 0.00010 | Hand trace of method logic indicates that the method return value matches expectations. Length is in between tolerance range. |
| 1.0005 | Meter stick is not within tolerance of 0.00010 | Meter stick is not within tolerance of 0.00010 | Hand trace of method logic indicates that the method return value matches expectations. Length is in between tolerance range. |

## Verifying method main

|  |  |  |  |
| --- | --- | --- | --- |
| Input | Expected return | Actual return | Description |
| 1  1 | Meter stick is within tolerance of 0.00010 | Meter stick is within tolerance of 0.00010 | Hand trace of program logic indicates that menu option for displaying verification report worked correctly. Numbers should be formatted to 5 decimal places |
| 1  2 | Centimeters: 100.00000 | Centimeters: 100.00000 | Hand trace of program logic indicates that menu option for displaying centimeters conversion worked correctly. |
| 1  0 | Invalid menu option selected. Please enter the correct option again: | Invalid menu option selected. Please enter the correct option again: | Hand trace of program logic indicates that program will output message to ask user enter the number again when the input is not a valid candidate number |
| 1  asd | Invalid menu option selected. Please enter the correct option again: | Invalid menu option selected. Please enter the correct option again: | Hand trace of program logic indicates that program will output message to ask user enter the number again when the input is a string |

# Part 4 Translate the Algorithm into Java

Graphical user interface, text, application

Description automatically generated

Figure . Class Exercise04 -1

Graphical user interface, text, application

Description automatically generated

Figure . Exercise04 - 2

Text

Description automatically generated

Figure . Class Exercise04 – 3

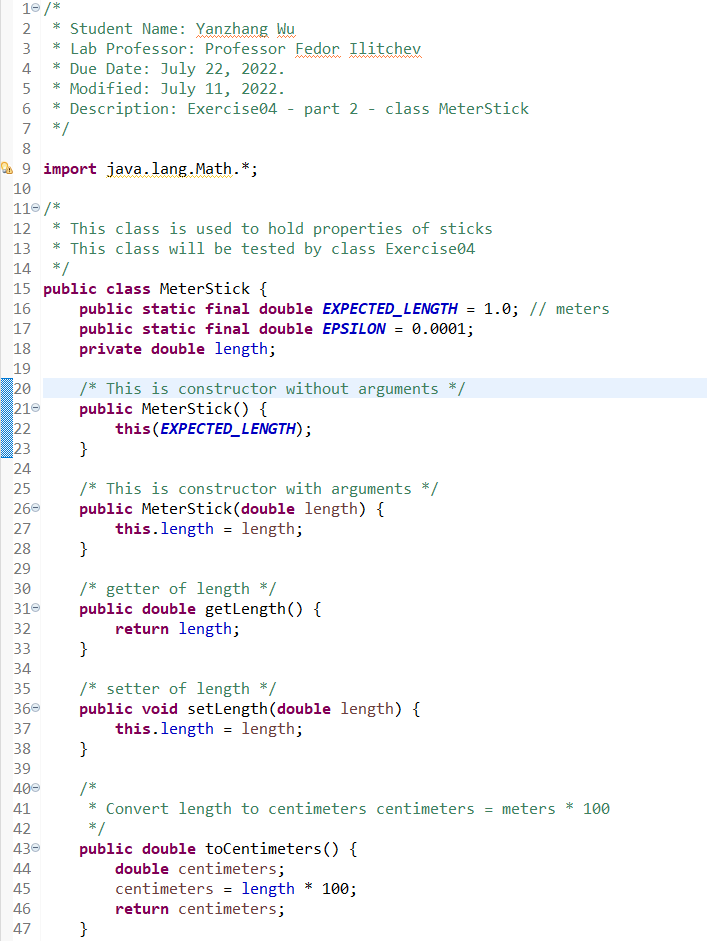


Figure . Class MeterStick – 1

Graphical user interface, text, application, email

Description automatically generated

Figure . Class MeterStick – 2

# Part 5 Test of the program

|  |  |  |  |
| --- | --- | --- | --- |
| Input | Expected return | Actual return | Description |
| asd | Invalid input. Please enter the correct length: | Invalid input. Please enter the correct length: | Java  program  output  matches the  expected  output |
| 0 | Invalid input. Please enter the correct length: | Invalid input. Please enter the correct length: | Java  program  output  matches the  expected  output |
| -5 | Invalid input. Please enter the correct length: | Invalid input. Please enter the correct length: | Java  program  output  matches the  expected  output |
| 1.02  a | Meter stick checker program.  Enter measured length in meters: 1.02  Please select from one of the following:  1 to validate meter stick  2 to show meter stick length in centimeters  3 to show meter stick length in millimeters  4 to show meter stick report  a  Invalid menu option selected. Please enter the correct option again: | Meter stick checker program.  Enter measured length in meters: 1.02  Please select from one of the following:  1 to validate meter stick  2 to show meter stick length in centimeters  3 to show meter stick length in millimeters  4 to show meter stick report  a  Invalid menu option selected. Please enter the correct option again: | Java  program  output  matches the  expected  output |
| 1.02  -5 | Meter stick checker program.  Enter measured length in meters: 1.02  Please select from one of the following:  1 to validate meter stick  2 to show meter stick length in centimeters  3 to show meter stick length in millimeters  4 to show meter stick report  -5  Invalid menu option selected. Please enter the correct option again: | Meter stick checker program.  Enter measured length in meters: 1.02  Please select from one of the following:  1 to validate meter stick  2 to show meter stick length in centimeters  3 to show meter stick length in millimeters  4 to show meter stick report  -5  Invalid menu option selected. Please enter the correct option again: | Java  program  output  matches the  expected  output |
| 1.0001  1 | Meter stick checker program.  Enter measured length in meters: 1.0001  Please select from one of the following:  1 to validate meter stick  2 to show meter stick length in centimeters  3 to show meter stick length in millimeters  4 to show meter stick report  1  Meter stick is within tolerance of 0.00010  Program by Yanzhang Wu | Meter stick checker program.  Enter measured length in meters: 1.0001  Please select from one of the following:  1 to validate meter stick  2 to show meter stick length in centimeters  3 to show meter stick length in millimeters  4 to show meter stick report  1  Meter stick is within tolerance of 0.00010  Program by Yanzhang Wu | Java  program  output  matches the  expected  output |
| 1.0001  2 | Meter stick checker program.  Enter measured length in meters: 1.0001  Please select from one of the following:  1 to validate meter stick  2 to show meter stick length in centimeters  3 to show meter stick length in millimeters  4 to show meter stick report  2  Centimeters: 100.01000  Program by Yanzhang Wu | Meter stick checker program.  Enter measured length in meters: 1.0001  Please select from one of the following:  1 to validate meter stick  2 to show meter stick length in centimeters  3 to show meter stick length in millimeters  4 to show meter stick report  2  Centimeters: 100.01000  Program by Yanzhang Wu | Java  program  output  matches the  expected  output |
| 1.0001  3 | Meter stick checker program.  Enter measured length in meters: 1.0001  Please select from one of the following:  1 to validate meter stick  2 to show meter stick length in centimeters  3 to show meter stick length in millimeters  4 to show meter stick report  3  Millimeters: 1000.10000  Program by Yanzhang Wu | Meter stick checker program.  Enter measured length in meters: 1.0001  Please select from one of the following:  1 to validate meter stick  2 to show meter stick length in centimeters  3 to show meter stick length in millimeters  4 to show meter stick report  3  Millimeters: 1000.10000  Program by Yanzhang Wu | Java  program  output  matches the  expected  output |
| 1.0001  4 | Meter stick checker program.  Enter measured length in meters: 1.0001  Please select from one of the following:  1 to validate meter stick  2 to show meter stick length in centimeters  3 to show meter stick length in millimeters  4 to show meter stick report  4  MeterStick length adjacent 1.00010,  Program by Yanzhang Wu | Meter stick checker program.  Enter measured length in meters: 1.0001  Please select from one of the following:  1 to validate meter stick  2 to show meter stick length in centimeters  3 to show meter stick length in millimeters  4 to show meter stick report  4  MeterStick length adjacent 1.00010,  Program by Yanzhang Wu | Java  program  output  matches the  expected  output |
| 1.001  1 | Meter stick checker program.  Enter measured length in meters: 1.001  Please select from one of the following:  1 to validate meter stick  2 to show meter stick length in centimeters  3 to show meter stick length in millimeters  4 to show meter stick report  1  Meter stick is not within tolerance of 0.00010  Program by Yanzhang Wu | Meter stick checker program.  Enter measured length in meters: 1.001  Please select from one of the following:  1 to validate meter stick  2 to show meter stick length in centimeters  3 to show meter stick length in millimeters  4 to show meter stick report  1  Meter stick is not within tolerance of 0.00010  Program by Yanzhang Wu | Java  program  output  matches the  expected  output |
|  |  |  |  |
|  |  |  |  |

Text

Description automatically generated

Figure . Invalid input of length

Text

Description automatically generated

Figure . Invalid menu section – a

Text

Description automatically generated

Figure . Invalid menu section – -5

Text, letter

Description automatically generated

Figure . Input 1.0001 & 1

Text, letter

Description automatically generated

Figure . Input 1.0001 & 2

Text, letter

Description automatically generated

Figure . Input 1.0001 & 3

Text, letter

Description automatically generated

Figure . Input 1.0001 & 4

Text, letter

Description automatically generated

Figure . Input 1.001 & 1