

## CS139 Programming Fundamentals Activity 05B : Testing Methods

### Objectives

At the end of this activity, students will be able to:

- Review and use method terminology
- Given a method header and its description, define appropriate test cases
- Use team members in different roles.

### Part 1 – Warm-up – Applying terminology

Given the temperature program below, answer the questions that follow.

1.	/**
2.	* TemperatureConverter contains methods to convert temperatures.
3.	* @author
4.	* @version
5.	*/
6.	public class TemperatureConverter
7.	{
8.	/**
9.	* main will exercise the two conversion methods
10.	* @param args the command line arguments
11.	*/
12.	public static void main(String[] args)
13.	{
14.	System.out.println("100 C s/b 212F is " + convertToFahrenheit(100.0));
15.	System.out.println("32 F s/b 0C is " + convertToCelcius(32.0));
16.	}
17.	/** convertToFahrenheit converts a celcius temperature to
18.	* Fahrenheit equivalent
19.	* @param celcius The celcius temperature to convert
20.	* @return the equivalent Fahrenheit temperature
21.	*/
22.	public static double convertToFahrenheit(double celcius)
23.	{
24.	double fahrenheit;
25.	
26.	fahrenheit = (celcius * 9.0 / 5.0) + 32.0;
27.	
28.	return fahrenheit;
29.	}
30.	/** convertToFahrenheit converts a celcius temperature to
31.	* Fahrenheit equivalent
32.	* @param fahrenheit The Fahrenheit temperature to convert
33.	* @return the equivalent celcius temperature
34.	*/
35.	public static double convertToCelcius(double fahrenheit)
36.	{
37.	double celcius;
38.	
39.	celcius = (fahrenheit - 32.0) * 5.0 / 9.0;
40.	
41.	return celcius;
42.	}
43.	}

- **Individually**, fill in the chart below.

Identify ONE line where you find a method header?		What is the name of the method?	
Identify ONE line where you find a parameter?		What is the name of the parameter?	
Identify ONE line where you find a return statement?		What is being returned?	
Identify ONE line where you find a call to a method in this file?		What is the name of the method?	
Identify ONE line where you find an argument?		What is the argument?	

- **Group**, compare answers.

## Part 2 – Thinking about testing

A value returning method has a set of one or more inputs and for each combination of inputs, an expected output. Becoming a programmer means learning how to test your code such that your program reliably produces the expected output.

Given a method with the following method documentation and header:

```
/**
 * Given a String parameter, return a new String with the
 * characters in the reverse order.
 * For example, given abc the method would return cba.
 *
 * @param text The text to reverse
 * @return The reversed String
 */
public static String reverse(String text)
```

- In your group, consider at least 4 **different** test cases that you would use to test your code. Write them in the chart below:

Argument value	Result value

- Looking at your test cases, what properties make each case different? In other words, except for the particular letters or symbols used, what makes one case different from another? In the chart below, indicate for each value from the chart above what makes that case different from the one before it.

Argument value	What are you testing for?

- **BOARD – Recorder, write the Argument and Result values on the board for your group. Presenter, be prepared to explain how your test cases are different from one another.**

**Part 3– A more complicated example**

A value returning method has a set of one or more inputs and for each combination of inputs, an expected output. Becoming a programmer means learning how to test your code such that your program reliably produces the expected output.

Given a method with the following method documentation and header:

```
/**
 * Produce the amount of tax on the given amount based on the following rules:
 *   If the taxType is 'X' or 'x' (exempt), then tax amount is 0.
 *   If the taxType is 'M' or 'm', then tax is 11% of the amount.
 *   If the taxType is 'F' or 'f', then tax is 2% of the amount.
 *   If the taxType is anything else, then tax is 5% of the amount.
 *
 * @param amount The amount of the sale
 * @param taxType A code for the type of items were purchased
 * @return The amount of tax
 */
public static double calculateTax(double amount, char taxType)
```

- **BOARD –** What is the minimum number of tests required to test each of the different cases?
- What are those cases and what are you testing for? (There may be more lines than you need)

Argument value	Expected value	What are you testing for?

- On the back of this page, write a main method to carry out 3 of the test cases you have defined above. This space provided for your main methods.