SOEN 6441

ADVANCED PROGRAMMING PRACTICES

DELIVERABLE 2

TEAM E

Pouria Pirian (40207625)
Sachin Prakash (40218678)
Nishant Saini (40195801)
Chandra Sagar Reddy Sangu (40234194)
Omer Sayem (40226505)
Kajal Sehrawat (40230025)

Instructor:

Prof. PANKAJ KAMTHAN

Department of Computer Science and Software Engineering
Gina Cody School of Engineering and Computer Science
Concordia University, Montreal

April 10, 2023

Contents

Overview

1.1 Introduction

We have received Source Code and documentation from team D (Detroit Cartel) for **CHEERS** problem. This include modules of the python program, Latex document, test cases for dynamic testing, Sample output file and read me file.

1.2 Objective

There are two primary objectives of Deliverable 2. They are as follows.

- To review source code systematically against best practices and style guideline that it claims to conform to.
- To test program P of Source Code S systematically using list of test cases.

Problem 4

Problem 5

3.1 Test Cases

The following test cases were executed.

3.1.1 Encoder - Test XML Response

This test case checks the generated XML file from the encoder module. The generated XML's structure and values are checked against the expected XML.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE result
  SYSTEM 'cheers.dtd'>
<result>
  <input>
    <alpha initial_guess="1.0">0.5</alpha>
  </input>
  <output>
    <record>
      <radius>2.0</radius>
      <length>3.0</length>
    </record>
    <record>
      <radius>4.0</radius>
      <length>5.0</length>
    </record>
    <record>
      <radius>6.0</radius>
      <length>7.0</length>
    </record>
  </output>
</result>
```

Figure 3.1: Expected XML

Encoder - Test XML Response	
Input	**Expected Output**
$initial\_guess = 1.0$ $alpha = 0.5$ $records = [(2.0,\ 3.0),\ (4.0,\ 5.0),\ (6.0,\ 7.0)]$	The encoder must generate an XML conforming to the DTD, with values matching the expected values.

3.1.2 Encoder - Test CSV Response

This test case checks the generated csv file from the encoder module. For every line in the CSV, the calculated value is checked against the expected value.

```
alpha,radius,length
0.5,2.0,3.0
0.5,4.0,5.0
0.5,6.0,7.0
```

Figure 3.2: Expected CSV

Encoder - Test CSV Response	
Input	**Expected Output**
alpha = 0.5 *records* = [(2.0, 3.0), (4.0, 5.0), (6.0, 7.0)]	The encoder must generate a CSV file matching the expected values.

3.1.3 Math - Test Sin

This test case checks the sin() function in the Math and libmath library/module. The Sin values returned from the Math library and libmath module have been compared.

Math - Test Sin	
Input	**Expected Output**
libmath.sin(0), math.sin (0)	Calculated values must be almost equal
libmath.sin(math.pi), math.sin(math.pi)	Calculated values must be almost equal
libmath.sin(math.pi/2), math.sin(math.pi/2)	Calculated values must be almost equal

3.1.4 Math - Test Cos

This test case checks the cos() function in the Math and libmath library/module. The Cos values returned from the Math library and libmath module have been compared.

Math - Test Cos	
Input	**Expected Output**
libmath.cos(0), math.cos(0)	Calculated values must be almost equal
libmath.cos(math.pi), math.cos(math.pi)	Calculated values must be almost equal
libmath.cos(math.pi/2), math.cos(math.pi/2)	Calculated values must be almost equal

3.1.5 Math - Test Pi

This test case checks the pi value in the Math library and the value_of_pi() method in the libmath module. The Pi value has been approximated 1000 times in the value_of_pi() method. The Pi values returned from the Math library and libmath module have been compared.

Math - Test Pi	
Input	**Expected Output**
libmath.value_of_pi(1000),math.pi, places = 2	Calculated values must be almost equal

3.1.6 Math - Test Factorial

This test case checks the factorial calculated by the libmath module. The calculated value is compared against the a few expected values.

Math - Test Factorial	
Input	**Expected Output**
libmath.factorial(0), 1	Calculated and expected value must match
libmath.factorial(1), 1	Calculated and expected value must match
libmath.factorial(5), 120	Calculated and expected value must match

3.1.7 Root Approximation - Test Numeric

This test case checks if an exception is thrown when a non-numeric epsilon (accuracy) value is passed to the newton_method() function.

Root Approximation - Test Numeric	
Input	**Expected Output**
newton_method(callable(), callable(), 'a')	A 'Value Error' exception must be raised.

3.1.8 Root Approximation - Test Positive

This test case checks if an exception is thrown when a negative epsilon (accuracy) value is passed to the newton_method() function.

Root Approximation - Test Positive	
Input	**Expected Output**
newton_method(callable(), callable(), '-0.01')	A 'Value Error' exception must be raised.

3.1.9 Root Approximation - Test Divide by Zero

This test case checks the callable lambda function passed to newton_method() function.

Root Approximation - Test Divide by Zero	
Input	**Expected Output**
newton_method(callable(), lambda x: 0, 1)	None must be returned.

3.1.10 Root Approximation - Error Tolerance

This test case checks if the error tolerance in the value calculated by newton_method() is less than 0.0001. Value for Sin, Pi and Cos have been obtained from the libmath module.

Root Approximation - Error Tolerance	
Input	**Expected Output**
newton_method(lambda a: a - sin(a) - PI/2, lambda a: 1 - cos(a), 1)	Calculated Value - 2.309878472457841 ¡ 0.0001

3.1.11 Execution Results

All the test cases passed the test and the code coverage reached 100%.

```
(venv) SachinscBookAir:soen-6441 sachinprakash$ pytest --cov -v
============================================================ test session starts ============================================================
platform darwin -- Python 3.11.1, pytest-7.2.2, pluggy-1.0.0 -- /Users/sachinprakash/vscode-workspace/urvil-app/soen-6441/venv/bin/python3.11
cachedir: .pytest_cache
rootdir: /Users/sachinprakash/vscode-workspace/urvil-app/soen-6441
plugins: cov-4.0.0
collected 10 items

tests/test_encoder.py::TestGenerateXMLResponse::test_generate_xml_response PASSED                                                    [ 10%]
tests/test_encoder.py::TestGenerateCSVResponse::test_generate_csv_response PASSED                                                    [ 20%]
tests/test_libmath.py::TestMathFunctions::test_cos PASSED                                                                            [ 30%]
tests/test_libmath.py::TestMathFunctions::test_factorial PASSED                                                                      [ 40%]
tests/test_libmath.py::TestMathFunctions::test_sin PASSED                                                                            [ 50%]
tests/test_libmath.py::TestMathFunctions::test_value_of_pi PASSED                                                                    [ 60%]
tests/test_root_approx.py::TestNewtonMethod::test_divide_by_zero PASSED                                                              [ 70%]
tests/test_root_approx.py::TestNewtonMethod::test_newton_method PASSED                                                               [ 80%]
tests/test_root_approx.py::TestNewtonMethod::test_numeric_x0 PASSED                                                                  [ 90%]
tests/test_root_approx.py::TestNewtonMethod::test_positive_epsilon PASSED                                                            [100%]

---------- coverage: platform darwin, python 3.11.1-final-0 ----------
Name                        Stmts   Miss  Cover
-----------------------------------------------
src/__init__.py                 0      0   100%
src/encoder.py                 35      0   100%
src/libmath.py                 25      0   100%
src/root_approx.py             17      0   100%
tests/__init__.py               0      0   100%
tests/test_encoder.py          45      0   100%
tests/test_libmath.py          18      0   100%
tests/test_root_approx.py      17      0   100%
-----------------------------------------------
TOTAL                         157      0   100%


============================================================ 10 passed in 0.15s =============================================================
(venv) SachinscBookAir:soen-6441 sachinprakash$
```

Figure 3.3: Test Results