Team International
Deliverable #2
Project: Celestia
09.26.2016
Members: Gui Costa, Megan Landau, Tony Tang

Task: Produce a detailed test plan of five test cases.

Report:

The following are detailed test plans for our current project, Celestia. We have specified 5 eventual test cases that we have developed for our software. Some of our test cases have already been tested, and others are still being built for testing. In order to be concise we have a detailed template for each test. We have a Test Suite ID, which at the moment is TS001; Test Case IDs which correspond to the test number; a Test Case Summary; Requirements; Components Being Tested; Methods Being Tested; Driver Being Tested; Test Inputs; Expected Outcomes; Prerequisites; Test Procedures; Test Data; Expected Results; Actual Results; Status (pass or fail); Remarks; Created By; and finally the Date Of Creation. Each field mentioned above was carefully recorded with information pertaining to its test.

Useful links:

http://softwaretestingfundamentals.com/test-case/
https://github.com/bgodard/celestia-g2/blob/89412cd52964b00e7b5429078304901fa848fec2/celestia/src/celmath/mathlib.h

Celestia Application Test Cases: Test Suite 001

Test Suite ID	TS001
Test Case ID	TC001
Test Case Summary	This test case will compute the output of the square method located in the mathlib.h library of the celestia-g2 project found on github
Requirement being tested	Test the output of the square method
Component being tested	Square method found in mathlib.h library
Method being tested	square(T x)
Driver being tested	testDriverSquare.cpp
Test input(s) including command-line arguments	2

Expected outcomes	4
Prerequisites	Mathlib.h must be found in the same directory as the test driver
Test Procedure	● First, get the mathlib.h file from the celestia-g2 repository found on github ● Next, create testDriverSquare.cpp for the test case driver. ● Make sure the mathlib.h file and .cpp file is in the same directory ● Type g++ test001 -o testDriverSquare.cpp into the terminal in Linux Ubuntu 16.04 ● Type ./test001 to execute the file ● Check to see if the expected outcome matches the actual outcome ● Record results
Test Data	https://github.com/bgodard/celestia-g2/blob/89412cd52964b00e7b5429078304901fa848fec2/celestia/src/celmath/mathlib.h
Expected Result	4
Actual Result	4
Status	Pass
Remarks	Test executed correctly, a was found to be 2 and the square of a is 4. The expected outcome was 4. So, the test passed.
Created by	Gui, Megan, Tony
Date of Creation	September 23, 2016 5:18 pm
Executed by	Gui, Megan, Tony
Date of Execution	September 22, 2016 12:50 pm
Test environment	OS: Linux Ubuntu 16.04

Test Suite ID	TS001
Test Case ID	TC002
Test Case Summary	Testing that the method can properly determine the mathematical outcome when finding area of a circle.
Requirement being tested	Test the output of the circleArea method
Component being tested	circleArea method found in mathlib.h library

Method being tested	circleArea(T r)
Test driver	testDriverCircleArea.cpp
Test inputs including command-line arguments	10
Prerequisites	Mathlib.h must be found in the same directory as the test driver
Test Procedure	● First, get the mathlib.h file from the celestia-g2 repository found on github ● Next, create testDriverCircleArea.cpp for the test case driver. ● Make sure the mathlib.h file and .cpp file is in the same directory ● Type g++ test002 -o testDriverCircleArea.cpp into the terminal in Linux Ubuntu 16.04 ● Type ./test002 to execute the file ● Check to see if the expected outcome matches the actual outcome ● Record results
Test Data	https://github.com/bgodard/celestia-g2/blob/89412cd52964b00e7b5429078304901fa848fec2/celestia/src/celmath/mathlib.h
Expected Result	314.15926...
Actual Result	314.159
Status	Pass
Remarks	The method rounded two decimals.
Created by	Gui, Megan, Tony
Date of Creation	September 23, 2016 5:18 pm
Executed by	Gui, Megan, Tony
Date of Execution	September 23, 2016 6:11 pm
Test environment	OS: Linux Ubuntu 16.04

Test Suite ID	TS001
Test Case ID	TC003
Test Case Summary	Testing the cube method found in the mathlib.h library.

Requirement being tested	Test the output of the cube method
Component being tested	Use a test driver program created in C++ to test the given method
Method being tested	cube (T x)
Test driver	testDriverCube.cpp
Test inputs including command-line arguments	2
Prerequisites	Mathlib.h must be found in the same directory as the test driver
Test Procedure	● First, get the mathlib.h file from the celestia-g2 repository found on github ● Next, create testDriverCube.cpp for the test case driver. ● Make sure the mathlib.h file and .cpp file is in the same directory ● Type g++ test003 -o testDriverCube.cpp into the terminal in Linux Ubuntu 16.04 ● Type ./test003 to execute the file ● Check to see if the expected outcome matches the actual outcome ● Record results
Test Data	https://github.com/bgodard/celestia-g2/blob/89412cd52964b00e7b5429078304901fa848fec2/celestia/src/celmath/mathlib.h
Expected Result	8
Actual Result	8
Status	Pass
Remarks	Output resulted in 8 which matched the predicted outcome. This method works similarly to the square method.
Created by	Gui, Megan, Tony
Date of Creation	September 23, 2016 5:18 pm
Executed by	Gui, Megan, Tony
Date of Execution	September 23, 2016 5:49 pm
Test environment	OS: Linux Ubuntu 16.04

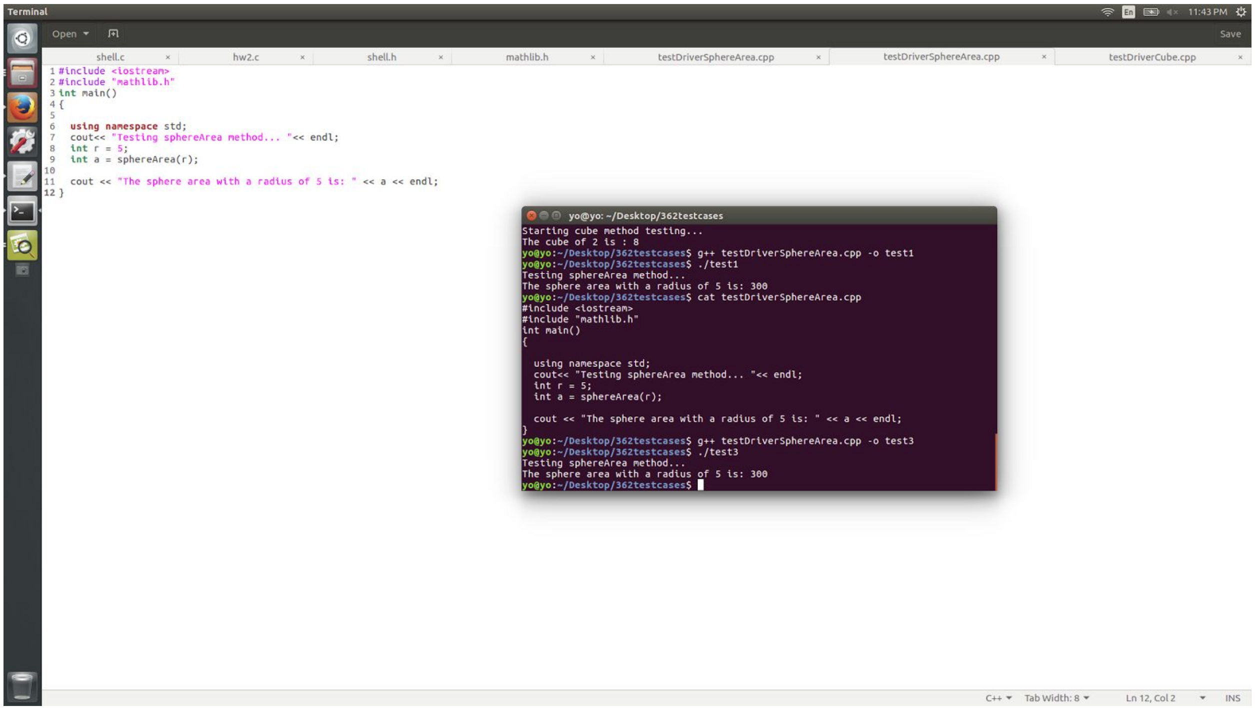
Test Suite ID	TS001

Test Case ID	TC004
Test Case Summary	Testing the sphereArea method found in the mathlib.h library
Requirement being tested	Test the output of the sphereArea method
Component being tested	Use a test driver program created in C++ to test the given method
Method being tested	sphereArea(T r)
Test driver	testDriverSphereArea.cpp
Test inputs including command-line arguments	5
Prerequisites	1. Mathlib.h must be found in the same directory as the test driver
Test Procedure	● First, get the mathlib.h file from the celestia-g2 repository found on github ● Next, create testDriverSphereArea.cpp for the test case driver. ● Make sure the mathlib.h file and .cpp file is in the same directory ● Type g++ test004 -o testDriverSphereArea.cpp into the terminal in Linux Ubuntu 16.04 ● Type ./test004 to execute the file ● Check to see if the expected outcome matches the actual outcome ● Record results
Test Data	https://github.com/bgodard/celestia-g2/blob/89412cd52964b00e7b5429078304901fa848fec2/celestia/src/celmath/mathlib.h
Expected Result	314
Actual Result	310
Status	Pass
Remarks	Found the output for the surface area of a sphere. The expected result matched the actual result.
Created by	Gui, Megan, Tony
Date of Creation	September 23, 2016 6:20 pm
Executed by	Gui, Megan, Tony
Date of Execution	September 23, 2016 7:58 pm
Test environment	OS: Linux Ubuntu 16.04

Test Suite ID	TS001
Test Case ID	TC005
Test Case Summary	Testing the readChar() method to ensure that it reads input stream and converts it to a char correctly
Requirement being tested	Char conversion and recognizability, exception handling
Component being tested	3dsread.cpp
Method being tested	readChar()
Test driver	testDriverReadChar.cpp
Test inputs including command-line arguments	char t
Prerequisites	1. 3dsread.cpp file has compiled and ran
Test Procedure	● First, get the 3dsread.cpp file from the celestia-g2 repository found on github ● Next, create testDriverReadChar.cpp for the test case driver. ● Make sure the 3dsread.cpp file is in the same directory ● Type g++ test005 -o testDriverReadChar.cpp into the terminal in Linux Ubuntu 16.04 ● Type ./test005 to execute the file ● Check to see if the expected outcome matches the actual outcome ● Record results
Test Data	https://github.com/bgodard/celestia-g2/blob/master/celestia/src/cel3ds/3dsread.cpp
Expected Result	t
Actual Result	----- (not yet recorded)
Status	----- (pass/fail not yet recorded)
Remarks	-----
Created by	Gui, Megan, Tony
Date of Creation	September 23, 2016 6:21 pm
Executed by	Gui, Megan, Tony
Date of Execution	-----
Test environment	OS: Linux Ubuntu 16.04

Screenshots:

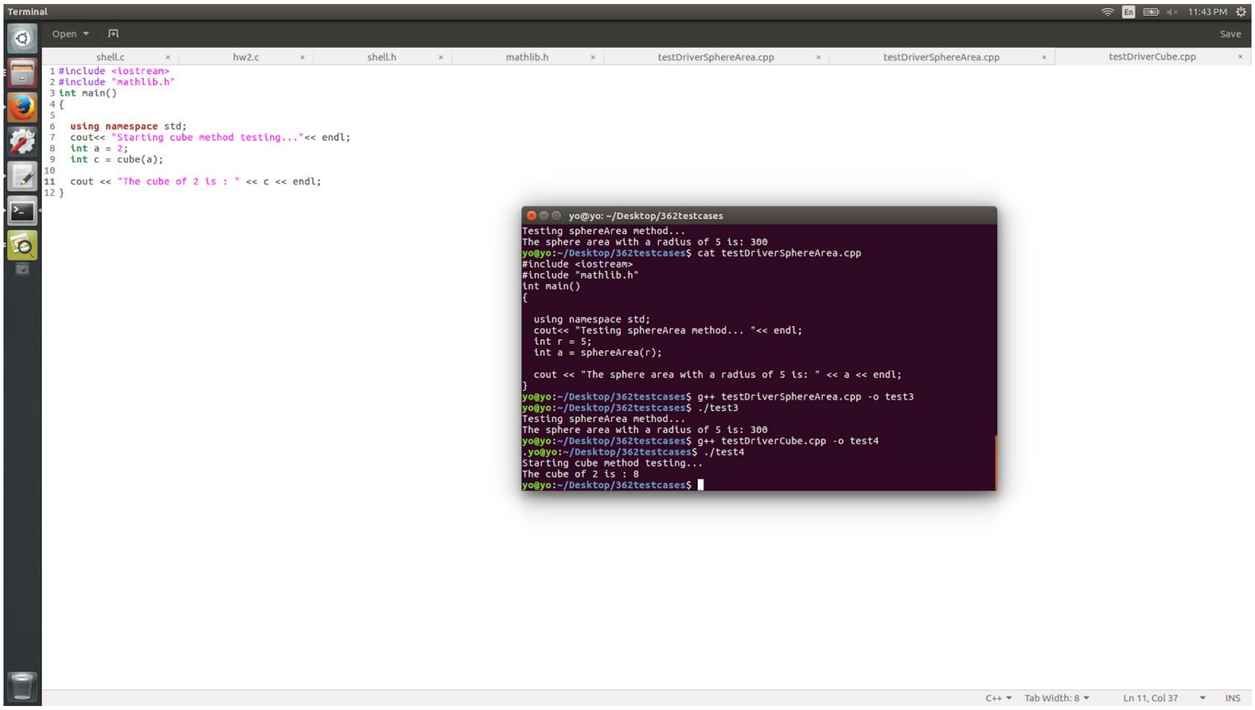
Testing the sphereArea() method from mathlib.h:



```
#include <iostream>
#include "mathlib.h"
int main()
{

  using namespace std;
  cout<< "Testing sphereArea method... "<< endl;
  int r = 5;
  int a = sphereArea(r);

  cout << "The sphere area with a radius of 5 is: " << a << endl;
}
```

```
Starting cube method testing...
The cube of 2 is : 8
yo@yo:~/Desktop/362testcases$ g++ testDriverSphereArea.cpp -o test1
yo@yo:~/Desktop/362testcases$ ./test1
Testing sphereArea method...
The sphere area with a radius of 5 is: 300
yo@yo:~/Desktop/362testcases$ cat testDriverSphereArea.cpp
#include <iostream>
#include "mathlib.h"
int main()
{

  using namespace std;
  cout<< "Testing sphereArea method... "<< endl;
  int r = 5;
  int a = sphereArea(r);

  cout << "The sphere area with a radius of 5 is: " << a << endl;
}
yo@yo:~/Desktop/362testcases$ g++ testDriverSphereArea.cpp -o test3
yo@yo:~/Desktop/362testcases$ ./test3
Testing sphereArea method...
The sphere area with a radius of 5 is: 300
yo@yo:~/Desktop/362testcases$ 
```

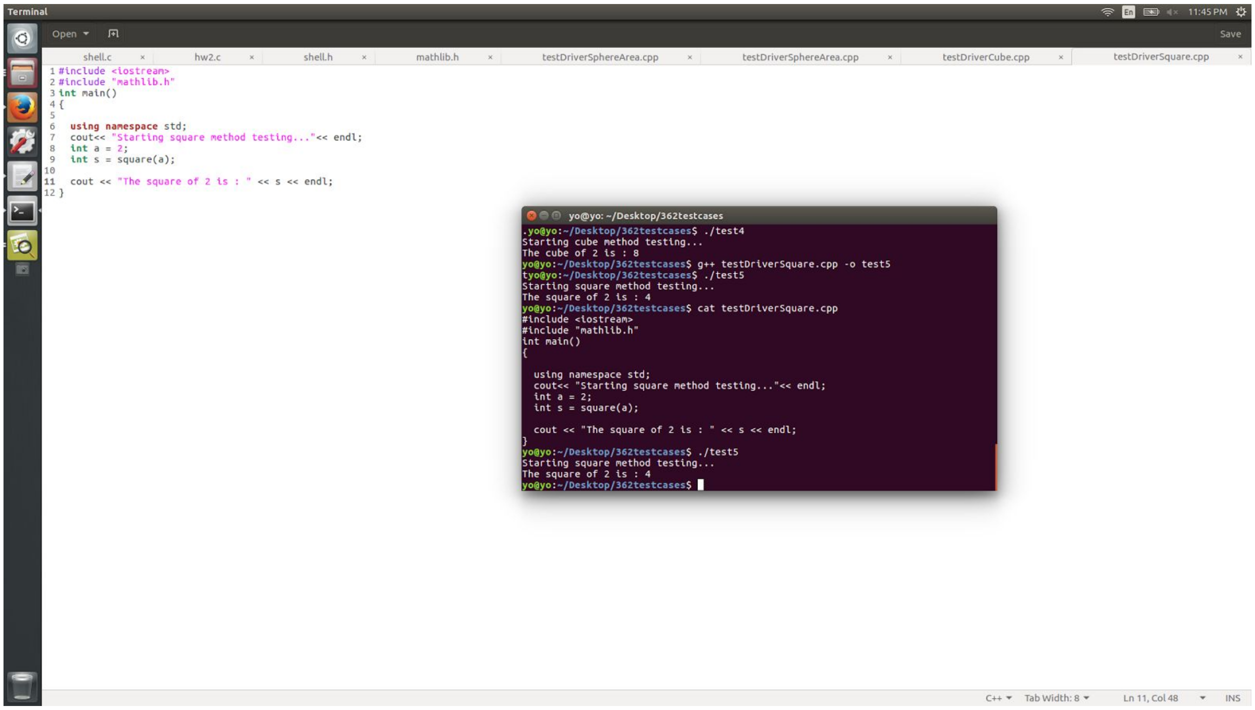
Testing the cube() method from mathlib.h:



```
#include <iostream>
#include "mathlib.h"
int main()
{

  using namespace std;
  cout<< "Starting cube method testing..."<< endl;
  int a = 2;
  int c = cube(a);

  cout << "The cube of 2 is : " << c << endl;
}
```

```
Testing sphereArea method...
The sphere area with a radius of 5 is: 300
yo@yo:~/Desktop/362testcases$ cat testDriverSphereArea.cpp
#include <iostream>
#include "mathlib.h"
int main()
{

  using namespace std;
  cout<< "Testing sphereArea method... "<< endl;
  int r = 5;
  int a = sphereArea(r);

  cout << "The sphere area with a radius of 5 is: " << a << endl;
}
yo@yo:~/Desktop/362testcases$ g++ testDriverSphereArea.cpp -o test3
yo@yo:~/Desktop/362testcases$ ./test3
Testing sphereArea method...
The sphere area with a radius of 5 is: 300
yo@yo:~/Desktop/362testcases$ g++ testDriverCube.cpp -o test4
.yo@yo:~/Desktop/362testcases$ ./test4
Starting cube method testing...
The cube of 2 is : 8
yo@yo:~/Desktop/362testcases$ 
```

Testing the square() method from mathlib.h:



```
#include <iostream>
#include "mathlib.h"
int main()
{

  using namespace std;
  cout<< "Starting square method testing..."<< endl;
  int a = 2;
  int s = square(a);

  cout << "The square of 2 is : " << s << endl;
}
```

```
.yo@yo:~/Desktop/362testcases$ ./test4
Starting cube method testing...
The cube of 2 is : 8
yo@yo:~/Desktop/362testcases$ g++ testDriverSquare.cpp -o test5
tyo@yo:~/Desktop/362testcases$ ./test5
Starting square method testing...
The square of 2 is : 4
yo@yo:~/Desktop/362testcases$ cat testDriverSquare.cpp
#include <iostream>
#include "mathlib.h"
int main()
{

  using namespace std;
  cout<< "Starting square method testing..."<< endl;
  int a = 2;
  int s = square(a);

  cout << "The square of 2 is : " << s << endl;
}
yo@yo:~/Desktop/362testcases$ ./test5
Starting square method testing...
The square of 2 is : 4
yo@yo:~/Desktop/362testcases$
```