By Ruben van Breda

May 2020

UnitTesting with CUnit

Assignment 3

COMP10050

19200704

Project overview:

Unit testing

Individual program units are tested.

Testing functionality of methods

Max Tests

Test Cases Functions	Definition
test_max	random sets of data
test_max_zero	sets of 0s
test_max_positive	sets of positive values
test_max_negative	sets of negative values
test_max_all_same	sets of identical values

Running Test Cases For 'max' Function

```
Suite: MAX_TEST
  Test: max_fun ...FAILED
    1. /Users/rubenvanbreda/OneDrive/UCD/Software Engineering Project/UnitTesting/Max_Tests.c:40  - CU_ASSERT_EQUAL(max(arr2,3),-1)
    2. /Users/rubenvanbreda/OneDrive/UCD/Software Engineering Project/UnitTesting/Max_Tests.c:41  - CU_ASSERT_EQUAL(max(arr3,2),0)
  Test: max_positive ...FAILED
    1. /Users/rubenvanbreda/OneDrive/UCD/Software Engineering Project/UnitTesting/Max_Tests.c:79  - CU_ASSERT_EQUAL(max(arr3, 4),9)
  Test: max_negative ...FAILED
    1. /Users/rubenvanbreda/OneDrive/UCD/Software Engineering Project/UnitTesting/Max_Tests.c:100  - CU_ASSERT_EQUAL(max(arr,3),0)
    2. /Users/rubenvanbreda/OneDrive/UCD/Software Engineering Project/UnitTesting/Max_Tests.c:101  - CU_ASSERT_EQUAL(max(arr2,3),-10)
    3. /Users/rubenvanbreda/OneDrive/UCD/Software Engineering Project/UnitTesting/Max_Tests.c:102  - CU_ASSERT_EQUAL(max(arr3,2),0)
    4. /Users/rubenvanbreda/OneDrive/UCD/Software Engineering Project/UnitTesting/Max_Tests.c:103  - CU_ASSERT_EQUAL(max(arr4,8),-1.1)
  Test: max_zero ...FAILED
    1. /Users/rubenvanbreda/OneDrive/UCD/Software Engineering Project/UnitTesting/Max_Tests.c:58  - CU_ASSERT_EQUAL(max(arr,1),0)
    2. /Users/rubenvanbreda/OneDrive/UCD/Software Engineering Project/UnitTesting/Max_Tests.c:59  - CU_ASSERT_EQUAL(max(arr2,4),0)
  Test: max_all_same ...FAILED
    1. /Users/rubenvanbreda/OneDrive/UCD/Software Engineering Project/UnitTesting/Max_Tests.c:122  - CU_ASSERT_EQUAL(max(arr4,6),-1)

Run Summary:    Type  Total    Ran Passed Failed Inactive
              suites      1      1    n/a      0        0
               tests      5      5      0      5        0
             asserts     20     20     10     10      n/a

Elapsed time =    0.000 seconds
```

Error with 'max' function:

When the program enters the function the variable of type double, max is declared however not initilized to a value, this results in arbriatary numbers being assigned to max (see Figure 1). Leading to problems where the max is assigned a random number

that is greater than any element in the passed in array as input. Resulting in failures in the varous test cases.

Figure 1

```
double max (double array[], int size){  array:
    double max;  max: 6.95312677993179 41E-310
    for(int i =0; i < size ; i++){
        if(max < array[i])
            max = array[i];
    }
    return max;
```

To solve this issue we assign max the value of the first element of the array(see Figure 2), And we can see if we run our test casses again, but with our new code adjustments to solve the problem, Our solution will pass our various test cases (see Fiure 3)

Figure 2

```
double max_solution(double array[], int size){
    double max = array[0];
    for(int i =0; i < size ; i++){
        if(max < array[i])
            max = array[i];
    }
    return max;
```

Figure 3. Running the same test with the new code adjustments

```
Suite: MAX_TEST_SOLUTION
  Test: max_fun ...passed
  Test: max_positive ...passed
  Test: max_negative ...passed
  Test: max_zero ...passed
  Test: max_all_same ...passed

Run Summary:    Type  Total    Ran Passed Failed Inactive
              suites      1      1    n/a      0        0
               tests      5      5      5      0        0
             asserts     20     20     20      0      n/a

Elapsed time =    0.000 seconds
```

Average Tests

Test Case Functions	Definition
test_average	random sets of data
test_average_zero	sets of 0s
test_average_positive	sets of positive values
test_average_negative	sets of negative values
test_average_all_same	sets of identical values

Running Test Cases For ‘average’ Function

```
Suite: AVERAGE_TEST
  Test: average_fun_test ...FAILED
    1. /Users/rubenvanbreda/OneDrive/UCD/Software Engineering Project/UnitTesting/Average_Tests.c:39  - CU_ASSERT_EQUAL(average(arr,3),2)
    2. /Users/rubenvanbreda/OneDrive/UCD/Software Engineering Project/UnitTesting/Average_Tests.c:40  - CU_ASSERT_EQUAL(average(arr2,3),(4.0/3.0))
    3. /Users/rubenvanbreda/OneDrive/UCD/Software Engineering Project/UnitTesting/Average_Tests.c:41  - CU_ASSERT_EQUAL(average(arr3,4),1.6)
    4. /Users/rubenvanbreda/OneDrive/UCD/Software Engineering Project/UnitTesting/Average_Tests.c:42  - CU_ASSERT_EQUAL(average(arr4,4),545/2.0)
  Test: avg_positive ...FAILED
    1. /Users/rubenvanbreda/OneDrive/UCD/Software Engineering Project/UnitTesting/Average_Tests.c:129  - CU_ASSERT_EQUAL(average(arr,3),2.0)
    2. /Users/rubenvanbreda/OneDrive/UCD/Software Engineering Project/UnitTesting/Average_Tests.c:130  - CU_ASSERT_EQUAL(average(arr2,4),(2009.0/40))
    3. /Users/rubenvanbreda/OneDrive/UCD/Software Engineering Project/UnitTesting/Average_Tests.c:131  - CU_ASSERT_EQUAL(average(arr3,4),2)
  Test: avg_negative ...FAILED
    1. /Users/rubenvanbreda/OneDrive/UCD/Software Engineering Project/UnitTesting/Average_Tests.c:112  - CU_ASSERT_EQUAL(average(arr,3),-2.0)
    2. /Users/rubenvanbreda/OneDrive/UCD/Software Engineering Project/UnitTesting/Average_Tests.c:113  - CU_ASSERT_EQUAL(average(arr2,4),(-2009.0/40))
    3. /Users/rubenvanbreda/OneDrive/UCD/Software Engineering Project/UnitTesting/Average_Tests.c:114  - CU_ASSERT_EQUAL(average(arr3,4),-2)
  Test: avg_zero ...passed
  Test: avg_all_same ...FAILED
    1. /Users/rubenvanbreda/OneDrive/UCD/Software Engineering Project/UnitTesting/Average_Tests.c:93  - CU_ASSERT_EQUAL(average(arr,6),1)
    2. /Users/rubenvanbreda/OneDrive/UCD/Software Engineering Project/UnitTesting/Average_Tests.c:94  - CU_ASSERT_EQUAL(average(arr2,6),2)
    3. /Users/rubenvanbreda/OneDrive/UCD/Software Engineering Project/UnitTesting/Average_Tests.c:95  - CU_ASSERT_EQUAL(average(arr3,6),100)
    4. /Users/rubenvanbreda/OneDrive/UCD/Software Engineering Project/UnitTesting/Average_Tests.c:96  - CU_ASSERT_EQUAL(average(arr4,6),-1)
  Test: test_avg_increasing_size_by_one ...FAILED
    1. /Users/rubenvanbreda/OneDrive/UCD/Software Engineering Project/UnitTesting/Average_Tests.c:62  - CU_ASSERT_DOUBLE_EQUAL(average(arr,5),2.5,0)
    2. /Users/rubenvanbreda/OneDrive/UCD/Software Engineering Project/UnitTesting/Average_Tests.c:63  - CU_ASSERT_DOUBLE_EQUAL(average(arr2,5),3,0)
    3. /Users/rubenvanbreda/OneDrive/UCD/Software Engineering Project/UnitTesting/Average_Tests.c:64  - CU_ASSERT_DOUBLE_EQUAL(average(arr3,5),2.2,0)

Run Summary:    Type  Total    Ran Passed Failed Inactive
              suites      1      1    n/a      0        0
               tests      6      6      1      5        0
             asserts     19     19      2     17      n/a
```

Error with ‘average’ function:

Lets run through a cycle for the array [1,2,3]

Sum = 0

Size = 3

I = 0

I<Size-1:

    Sum = 1 +array[0] = 1

    I++

I=1

I<2:

    Sum = 3 +array[1] = 2

    I++

I=2

I!<2 break;

Sum = 3

Size = 3

Return Sum/Size -> 1 which is wrong

The problem is due to a under count of the elements being accumulated in the for loop (see Figure 4). This fix is simply done by removing the subrtaction of 1 from size in the for loop (see Figure 5)

Figure 4

```
double average(double array[], int size){
    double sum = 0;
    for(int i = 0; i < size-1; i++){
        sum += array[i];
    }
    return sum/size;
```

Figure 5

```
double average_solution(double array[], int size){
    double sum = 0;
    for(int i = 0; i < size; i++){
        sum += array[i];
    }
    return sum/size;
```

Running the tests cases with the average_solution function

```
Suite: AVERAGE_TEST_SOLUTION
  Test: average_fun_test ...passed
  Test: avg_positive ...passed
  Test: avg_negative ...passed
  Test: avg_zero ...passed
  Test: avg_all_same ...passed
  Test: test_avg_increasing_size_by_one ...passed

Run Summary:    Type  Total    Ran Passed Failed Inactive
              suites      1      1    n/a      0        0
               tests      6      6      6      0        0
             asserts     19     19     19      0      n/a

Elapsed time =    0.000 seconds
```