COMP1917: 13 Testing

Sim Mautner

s.mautner@unsw.edu.au

September 5, 2016

Assert

- Checks that a statement is true.
- Aborts the program if it is not true.
- The program continues running with no output if the statement is true.
- Ex 1: Write a program with the assert statements below and observe tests passing and failing.

```
#include <assert.h>

assert(1==1); // Passes

assert(1==2); // Fails
```

Test Driven Development (TDD)

- An approach to software development.
- Process:
  1. Read the specification.
  2. Think about what the code should do once it is written.
  3. Write tests which once the program is complete, will pass.
  4. Write the code.
  5. Run the tests to check that the code has been written correctly.

Testing and TDD

- Ex 1: Using Test Driven Development, write a function which sums the elements of an array.
  - Put function in a separate .c file from the tests.
  - Put the function prototype in a .h file.

1. What situations do we need to test?
2. Write the tests.
3. Write the code.

Make sure to test all different situations, pathways and boundaries of your code.

Testing and TDD

- Ex 2: Using Test Driven Development, write a function which sums all the odd integers in an array.
  - Put function in a separate .c file from the tests.
  - Put the function prototype in a .h file.

1. What situations do we need to test?
2. Write the tests.
3. Write the code.

Make sure to test all different situations, pathways and boundaries of your code.

Black Box Testing and White Box Testing

- Black Box Testing:
  - Doesn't know the "behind the scenes" implementation of the code.
  - Checks that the correct output occurs for a given input.
- White Box Testing:
  - Requires knowledge of and access to internal details of how the code works and the steps taken to achieve the goal.
  - Checks that these steps are carried out correctly.