– usage –

Your tests should be written as a single .c file separate from the body of text containing your functionality to be tested. A simple example might look something like this:

```c
/* keep this include at the very top of the file */
#include "simple-test.h"

/* any global variables, functions, other inclusions, etc.
 * should be declared here */
#include "header_with_stuff_to_be_tested.h"

BEGIN_TEST

/* the string is a description of the test being run */
TEST("check add()'s return value")
{
        int var1=2;
        int var2=4;

        /* add is a function included from our hypothetical
         * header_with_stuff_to_be_tested */
        EXPECT_INT(var1+var2, add(var1, var2));
}

TEST("compare two arrays of strings")
{
        int i;
        char array1[][10] = {
                "str1",
                "str2",
                "str3",
        };
        char array2[][10] = {
                "str1",
                "str2",
                /* matching will fail here */
                "different",
        };

        for(i = 0; i < sizeof(array1) / sizeof(char[10]); i++) {
                /* ECHO can be used to print (with pretty
                 * formatting) the current state within the
                 * test */
                ECHO("checking strs at i == %i", i);
                EXPECT_STR_EQ(array1[i], array2[i]);
        }
}

END_TEST
```

If the second test was ommited, the ouput would look like this:

```
a_simple_example.c
 1 :: check add()’s return value
   :: success!
```

If the second test was included, it would look like this:

```
a_simple_example.c
 1 :: check add()’s return value
 2 :: compare two arrays of strings
   :: checking strs at i == 0...
   :: checking strs at i == 1...
   :: checking strs at i == 2...
   :: a_simple_example.c:40: fail: strings unequal
   ::   array1[i] == ‘str3’
   ::   array2[i] == ‘different’
```

For detailed descriptions of all available tests, see the included man page or README.md .