Getting started with Google Test (GTest) on Ubuntu

BY ERIK SMISTAD · JULY 5, 2012

Google test is a framework for writing C++ unit tests. In this short post, I explain how to set it up in Ubuntu.

Start by installing the gtest development package:

```
sudo apt-get install libgtest-dev
```

Note that this package only install source files. You have to compile the code yourself to create the necessary library files. These source files should be located at /usr/src/gtest. Browse to this folder and use cmake to compile the library:

```
sudo apt-get install cmake # install cmake
cd /usr/src/gtest
sudo cmake CMakeLists.txt
sudo make

# copy or symlink libgtest.a and libgtest_main.a to your /usr/lib folder
sudo cp *.a /usr/lib
```

Lets say we now want to test the following simple squareRoot function:

```
// whattotest.cpp
#include <math.h>

double squareRoot(const double a) {
    double b = sqrt(a);
    if(b != b) { // nan check
        return -1.0;
    }else{
        return sqrt(a);
    }
}
```

In the following code, we create two tests that test the function using a simple assertion. There exists many other assertion macros in the framework (seehttp://code.google.com/p/googletest/wiki/Primer#Assertions). The code contains a small main function that will run all of the tests automatically. Nice and simple!

```
// tests.cpp
#include "whattotest.cpp"
#include <gtest/gtest.h>

TEST(SquareRootTest, PositiveNos) {
    ASSERT_EQ(6, squareRoot(36.0));
    ASSERT_EQ(18.0, squareRoot(324.0));
    ASSERT_EQ(25.4, squareRoot(645.16));
    ASSERT_EQ(0, squareRoot(0.0));
}

TEST(SquareRootTest, NegativeNos) {
    ASSERT_EQ(-1.0, squareRoot(-15.0));
    ASSERT_EQ(-1.0, squareRoot(-0.2));
}

int main(int argc, char **argv) {
    testing::InitGoogleTest(&argc, argv);
    return RUN_ALL_TESTS();
}
```

The next step is to compile the code. I've set up a small CMakeLists.txt file below to compile the tests. This file locates the google test library and links it with the test application. Note that we also have to link to the pthread library or the application won't compile.

```
cmake_minimum_required(VERSION 2.6)

# Locate GTest
find_package(GTest REQUIRED)
include_directories(${GTEST_INCLUDE_DIRS})

# Link runTests with what we want to test and the GTest and p
library
add_executable(runTests tests.cpp)
target_link_libraries(runTests ${GTEST_LIBRARIES} pthread
```

Compile and run the tests:

```
cmake CMakeLists.txt
make
./runTests
```

Have fun testing! You can download all of the code above at my Github page: https://github.com/smistad/GTest

References

http://code.google.com/p/googletest/wiki/Documentation
http://www.ibm.com/developerworks/aix/library/au-googletestingframework.html