UNIVERSIDAD NACIONAL DE SAN AGUSTÍN DE AREQUIPA

ESCUELA PROFESIONAL DE CIENCIA DE LA COMPUTACIÓN
INGENIERA DE SOFTWARE II

Laboratorio 3

Presentado por:
Fiorela Villarroel Ramos

Docente :
Edgar Sarmiento Calisaya

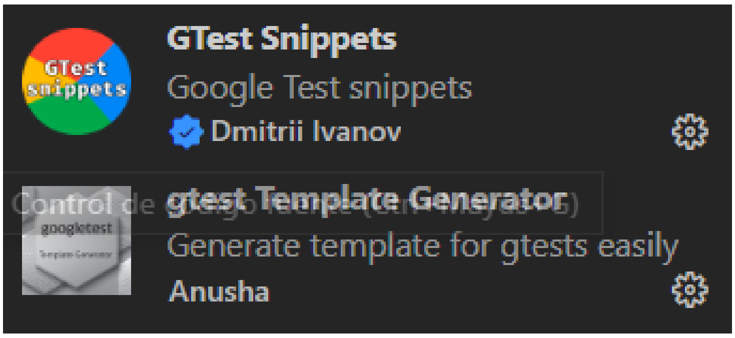
Laboratorio 4 : Pruebas Unitarias

1. Objetivo

Entender los conceptos fundamentales de pruebas unitarias en proyectos C++ usando el Framework GTest (Basado en xUnit)

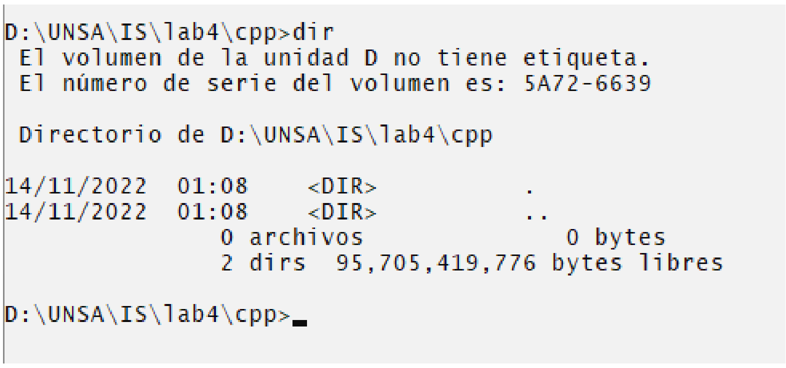
2. Actividades

1. Descargar el framework Google Test (GTest): Terminal o Visual Studio (`https://github.com/google/googletest`):



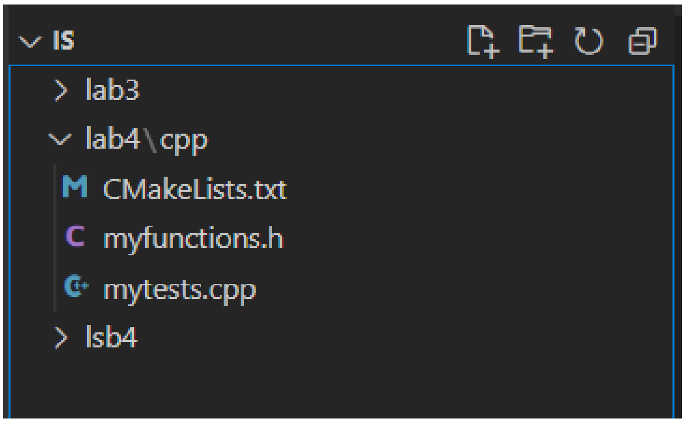
2. Construir (Build) GTest: Terminal o Visual Studio: Visual Studio Code: **Tutorial Visual Studio Code**

   a) Comience con un directorio limpio:

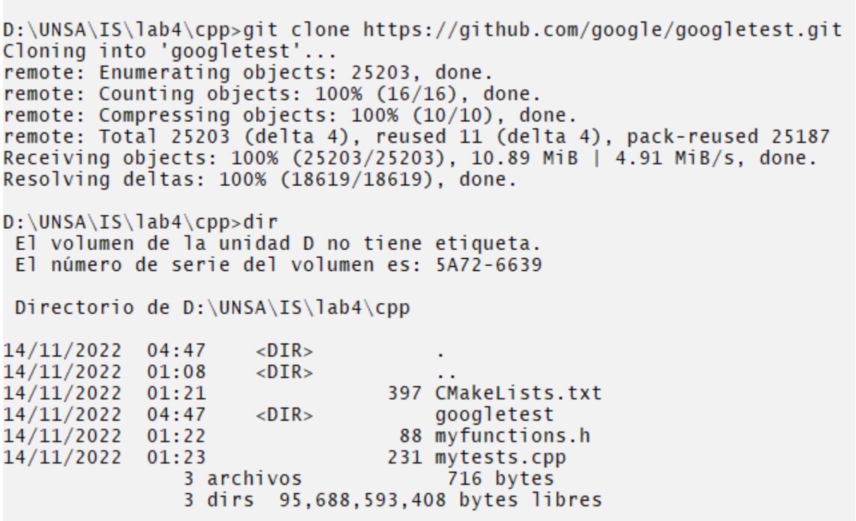


```
D:\UNSA\IS\lab4\cpp>dir
 El volumen de la unidad D no tiene etiqueta.
 El número de serie del volumen es: 5A72-6639

 Directorio de D:\UNSA\IS\lab4\cpp

14/11/2022  01:08    <DIR>          .
14/11/2022  01:08    <DIR>          ..
               0 archivos              0 bytes
               2 dirs  95,705,419,776 bytes libres

D:\UNSA\IS\lab4\cpp>
```

b) Agregue su archivo cmake (CMakeLists.txt), sus archivos fuente y el archivo de prueba. El directorio ahora se ve así:

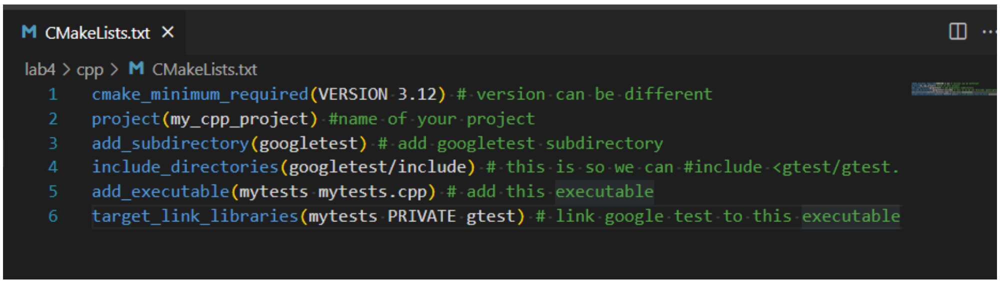


c) Clonar y agregar googletest a este directorio:



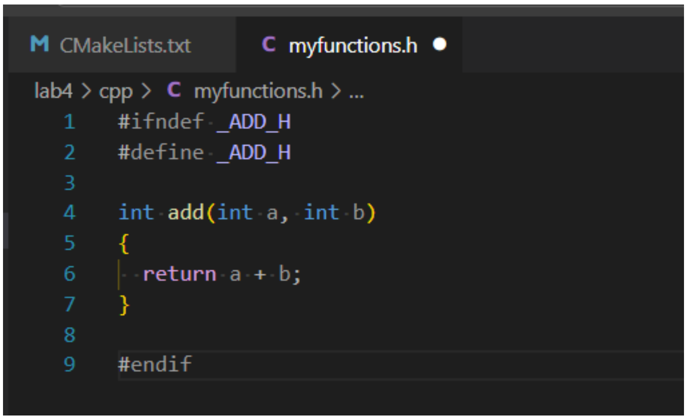
```
D:\UNSA\IS\lab4\cpp>git clone https://github.com/google/googletest.git
Cloning into 'googletest'...
remote: Enumerating objects: 25203, done.
remote: Counting objects: 100% (16/16), done.
remote: Compressing objects: 100% (10/10), done.
remote: Total 25203 (delta 4), reused 11 (delta 4), pack-reused 25187
Receiving objects: 100% (25203/25203), 10.89 MiB | 4.91 MiB/s, done.
Resolving deltas: 100% (18619/18619), done.

D:\UNSA\IS\lab4\cpp>dir
 El volumen de la unidad D no tiene etiqueta.
 El número de serie del volumen es: 5A72-6639

 Directorio de D:\UNSA\IS\lab4\cpp

14/11/2022  04:47    <DIR>          .
14/11/2022  01:08    <DIR>          ..
14/11/2022  01:21               397 CMakeLists.txt
14/11/2022  04:47    <DIR>          googletest
14/11/2022  01:22                88 myfunctions.h
14/11/2022  01:23               231 mytests.cpp
               3 archivos            716 bytes
               3 dirs  95,688,593,408 bytes libres
```

d) Abre tu CMakeLists.txt e ingresa lo siguiente:



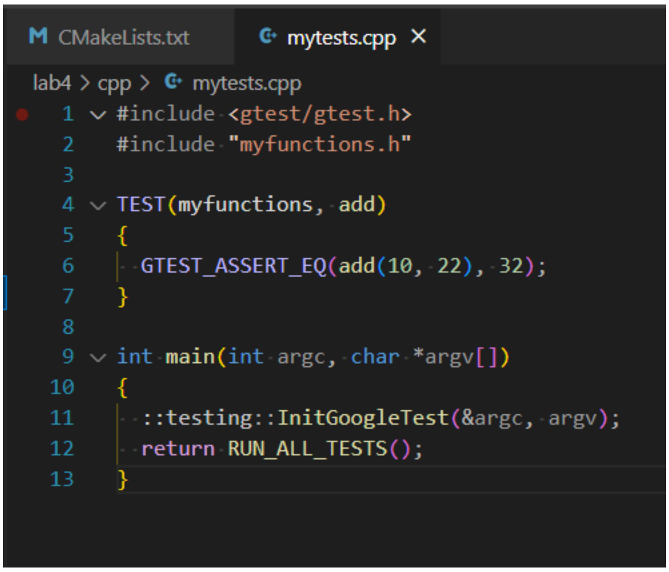
```
cmake_minimum_required(VERSION 3.12) # version can be different
project(my_cpp_project) #name of your project
add_subdirectory(googletest) # add googletest subdirectory
include_directories(googletest/include) # this is so we can #include <gtest/gtest.
add_executable(mytests mytests.cpp) # add this executable
target_link_libraries(mytests PRIVATE gtest) # link google test to this executable
```

e) Contenido de myfunctions.hpara el ejemplo:



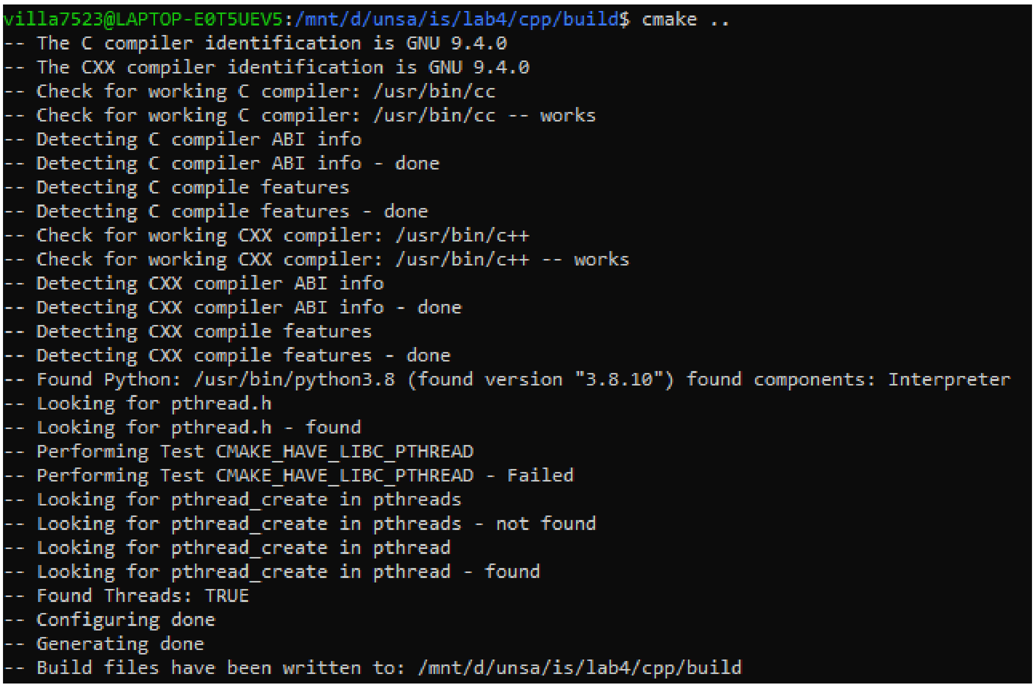
```cpp
#ifndef _ADD_H
#define _ADD_H

int add(int a, int b)
{
  return a + b;
}

#endif
```

f) Contenido de mytests.cpp para el ejemplo:

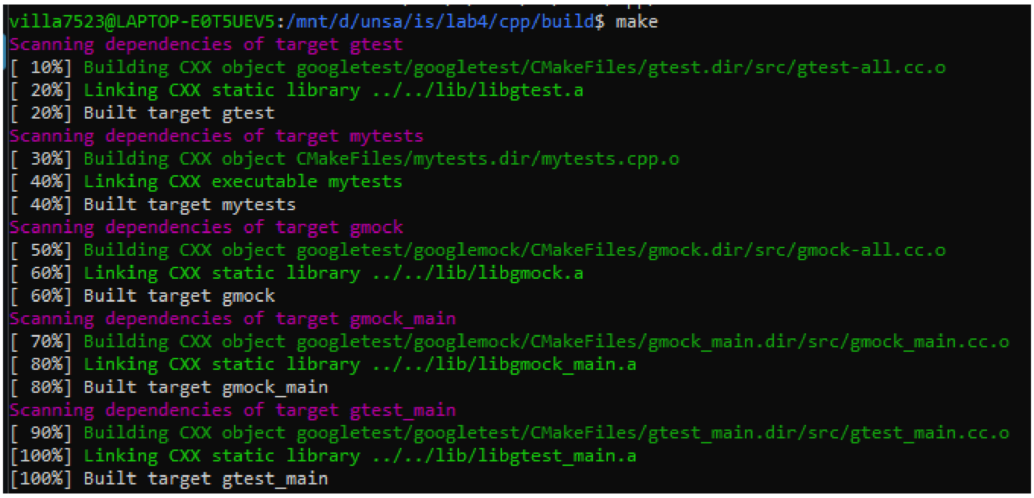


```cpp
#include <gtest/gtest.h>
#include "myfunctions.h"

TEST(myfunctions, add)
{
  GTEST_ASSERT_EQ(add(10, 22), 32);
}

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

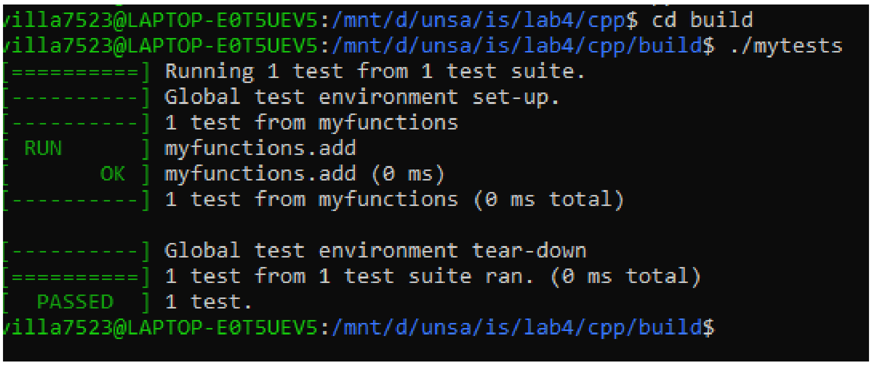
g) Finalmente realizar las pruebas:



```
villa7523@LAPTOP-E0T5UEV5:/mnt/d/unsa/is/lab4/cpp/build$ cmake ..
-- The C compiler identification is GNU 9.4.0
-- The CXX compiler identification is GNU 9.4.0
-- Check for working C compiler: /usr/bin/cc
-- Check for working C compiler: /usr/bin/cc -- works
-- Detecting C compiler ABI info
-- Detecting C compiler ABI info - done
-- Detecting C compile features
-- Detecting C compile features - done
-- Check for working CXX compiler: /usr/bin/c++
-- Check for working CXX compiler: /usr/bin/c++ -- works
-- Detecting CXX compiler ABI info
-- Detecting CXX compiler ABI info - done
-- Detecting CXX compile features
-- Detecting CXX compile features - done
-- Found Python: /usr/bin/python3.8 (found version "3.8.10") found components: Interpreter
-- Looking for pthread.h
-- Looking for pthread.h - found
-- Performing Test CMAKE_HAVE_LIBC_PTHREAD
-- Performing Test CMAKE_HAVE_LIBC_PTHREAD - Failed
-- Looking for pthread_create in pthreads
-- Looking for pthread_create in pthreads - not found
-- Looking for pthread_create in pthread
-- Looking for pthread_create in pthread - found
-- Found Threads: TRUE
-- Configuring done
-- Generating done
-- Build files have been written to: /mnt/d/unsa/is/lab4/cpp/build
```



```
villa7523@LAPTOP-E0T5UEV5:/mnt/d/unsa/is/lab4/cpp/build$ make
Scanning dependencies of target gtest
[ 10%] Building CXX object googletest/googletest/CMakeFiles/gtest.dir/src/gtest-all.cc.o
[ 20%] Linking CXX static library ../../lib/libgtest.a
[ 20%] Built target gtest
Scanning dependencies of target mytests
[ 30%] Building CXX object CMakeFiles/mytests.dir/mytests.cpp.o
[ 40%] Linking CXX executable mytests
[ 40%] Built target mytests
Scanning dependencies of target gmock
[ 50%] Building CXX object googletest/googlemock/CMakeFiles/gmock.dir/src/gmock-all.cc.o
[ 60%] Linking CXX static library ../../lib/libgmock.a
[ 60%] Built target gmock
Scanning dependencies of target gmock_main
[ 70%] Building CXX object googletest/googlemock/CMakeFiles/gmock_main.dir/src/gmock_main.cc.o
[ 80%] Linking CXX static library ../../lib/libgmock_main.a
[ 80%] Built target gmock_main
Scanning dependencies of target gtest_main
[ 90%] Building CXX object googletest/googletest/CMakeFiles/gtest_main.dir/src/gtest_main.cc.o
[100%] Linking CXX static library ../../lib/libgtest_main.a
[100%] Built target gtest_main
```



```
villa7523@LAPTOP-E0T5UEV5:/mnt/d/unsa/is/lab4/cpp$ cd build
villa7523@LAPTOP-E0T5UEV5:/mnt/d/unsa/is/lab4/cpp/build$ ./mytests
[==========] Running 1 test from 1 test suite.
[----------] Global test environment set-up.
[----------] 1 test from myfunctions
[ RUN      ] myfunctions.add
[       OK ] myfunctions.add (0 ms)
[----------] 1 test from myfunctions (0 ms total)

[----------] Global test environment tear-down
[==========] 1 test from 1 test suite ran. (0 ms total)
[  PASSED  ] 1 test.
villa7523@LAPTOP-E0T5UEV5:/mnt/d/unsa/is/lab4/cpp/build$
```

3. Crear un proyecto C++ (Sample): Escoger uno de: **GoogleTest** El proyecto escogido se encuentra en **Link**

4. Crear un proyecto de prueba unitaria usando Test Fixtures (SampleTest): ASSERT, EXPECT, TEST, TEST_F:

5. Diseñar e implementar 2 casos de prueba: Escoger de acuerdo a paso 4: **Sample Test**

    - Test de factorial de números negativos

    ```
    TEST(FactorialTest, Negative)
    {
      // Esta prueba es llamada "Negative", y comienza "FactorialTest"
      // Caso de prueba
      EXPECT_EQ(1, Factorial(-5));
      EXPECT_EQ(1, Factorial(-1));
      EXPECT_GT(Factorial(-10), 0);
    }
    ```

    - Test de factorial de zero

    ```
    TEST(FactorialTest, Zero) { EXPECT_EQ(1, Factorial(0)); }
    ```

    - Test factorial de números positivos

    ```
    TEST(FactorialTest, Positive)
    {
      EXPECT_EQ(1, Factorial(1));
      EXPECT_EQ(2, Factorial(2));
      EXPECT_EQ(6, Factorial(3));
      EXPECT_EQ(40320, Factorial(8));
    }
    ```

    - Test si un número negativo es primo

    ```
    TEST(IsPrimeTest, Negative)
      {
        // This test belongs to the IsPrimeTest test case.

        EXPECT_FALSE(IsPrime(-1));
        EXPECT_FALSE(IsPrime(-2));
    ```

```
    EXPECT_FALSE(IsPrime(INT_MIN));
  }
```

- Test de casos triviales

```
TEST(IsPrimeTest, Trivial)
  {
    EXPECT_FALSE(IsPrime(0));
    EXPECT_FALSE(IsPrime(1));
    EXPECT_TRUE(IsPrime(2));
    EXPECT_TRUE(IsPrime(3));
  }
```

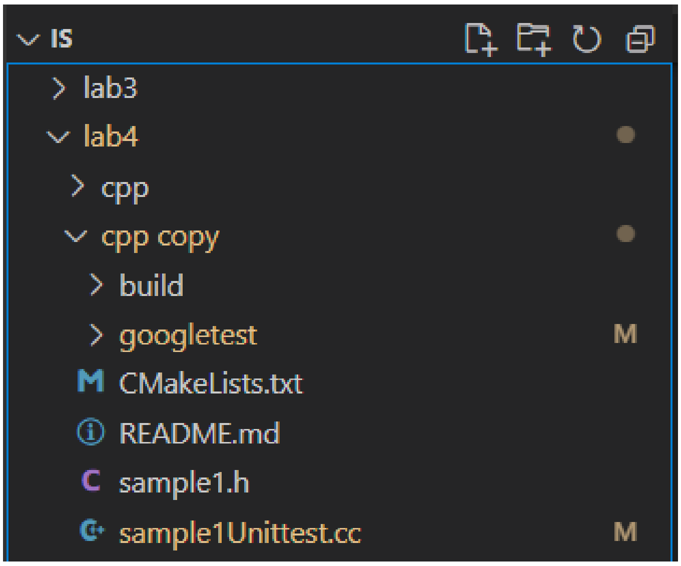
- Test de números positivos

```
TEST(IsPrimeTest, Positive)
  {
    EXPECT_FALSE(IsPrime(4));
    EXPECT_TRUE(IsPrime(5));
    EXPECT_FALSE(IsPrime(6));
    EXPECT_TRUE(IsPrime(23));
  }
```

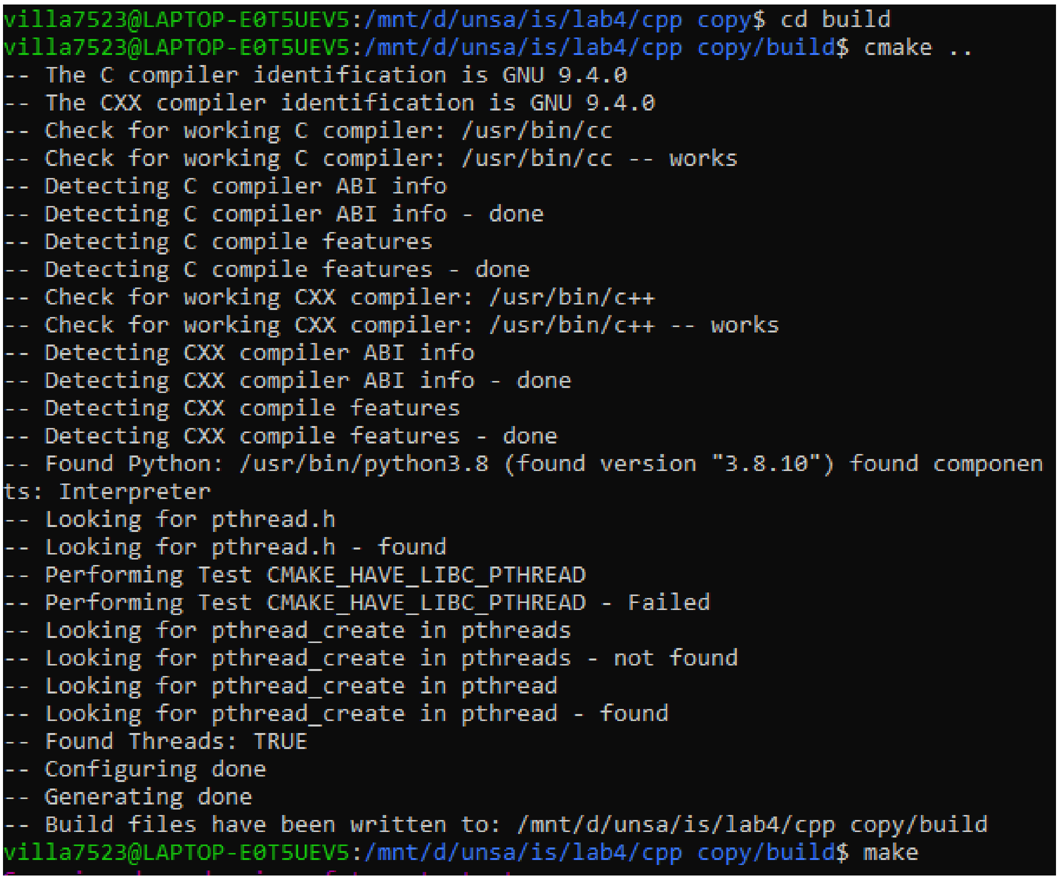
6. Reportar los resultados de ejecución de los casos de prueba (¿Pasó o Falló?): Por Terminal o Extension GTest/CMake en Visual Studio (Ver docs en paso 2):

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

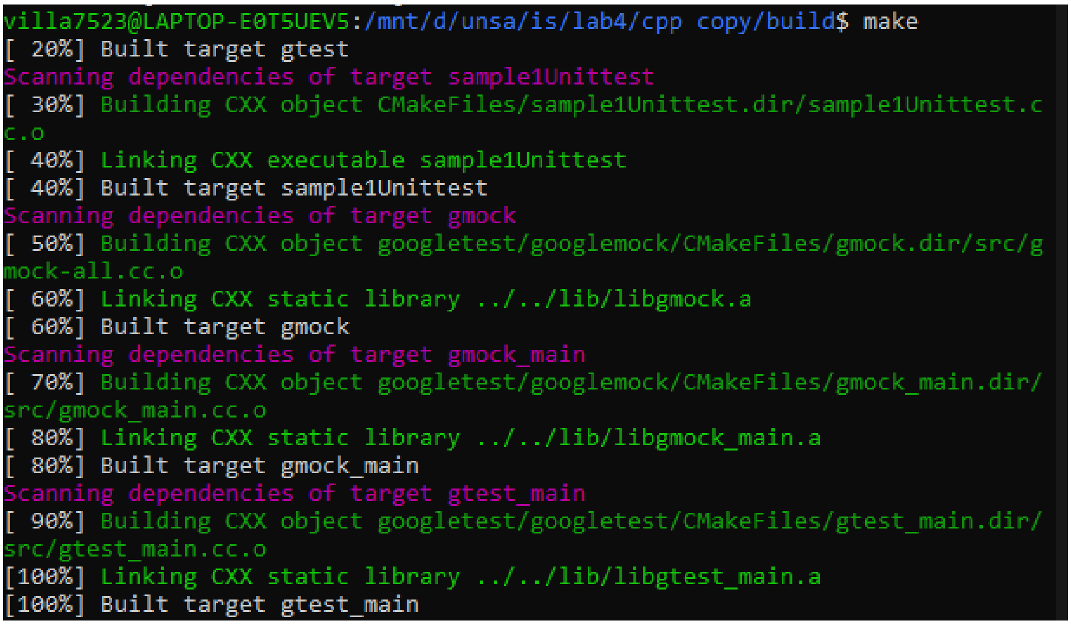
2.1. Resultados



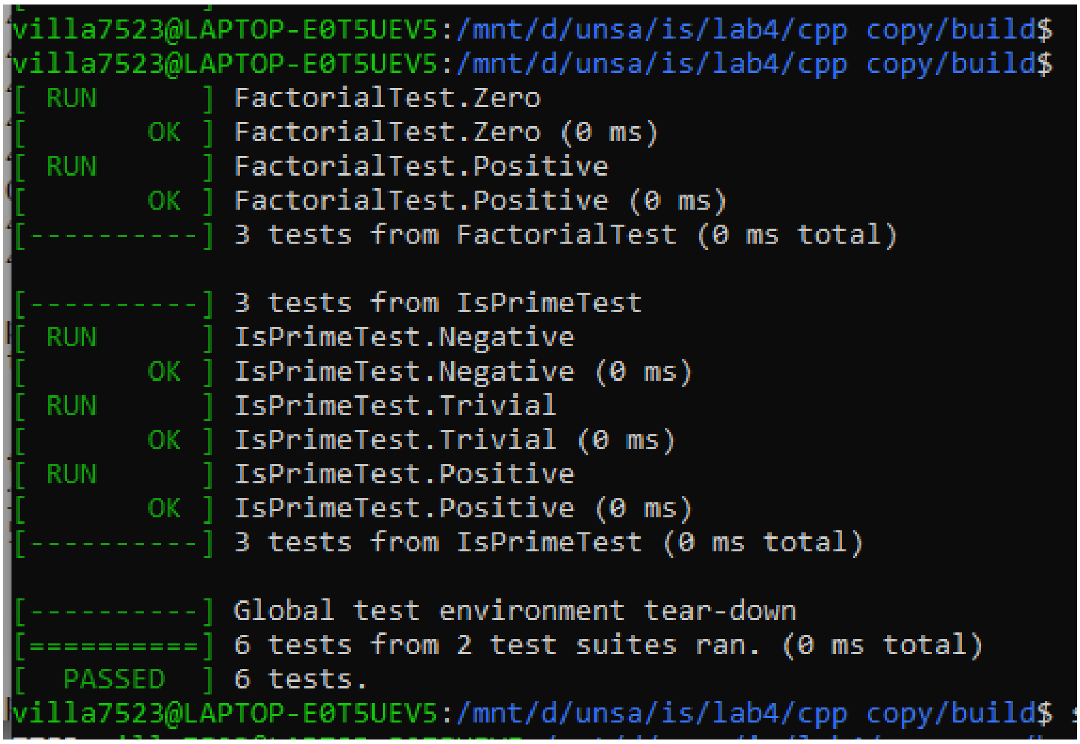
```
IS
  lab3
  lab4
    cpp
    cpp copy
      build
      googletest                M
      CMakeLists.txt
      README.md
      sample1.h
      sample1Unittest.cc        M
```



```
villa7523@LAPTOP-E0T5UEV5:/mnt/d/unsa/is/lab4/cpp copy$ cd build
villa7523@LAPTOP-E0T5UEV5:/mnt/d/unsa/is/lab4/cpp copy/build$ cmake ..
-- The C compiler identification is GNU 9.4.0
-- The CXX compiler identification is GNU 9.4.0
-- Check for working C compiler: /usr/bin/cc
-- Check for working C compiler: /usr/bin/cc -- works
-- Detecting C compiler ABI info
-- Detecting C compiler ABI info - done
-- Detecting C compile features
-- Detecting C compile features - done
-- Check for working CXX compiler: /usr/bin/c++
-- Check for working CXX compiler: /usr/bin/c++ -- works
-- Detecting CXX compiler ABI info
-- Detecting CXX compiler ABI info - done
-- Detecting CXX compile features
-- Detecting CXX compile features - done
-- Found Python: /usr/bin/python3.8 (found version "3.8.10") found componen
ts: Interpreter
-- Looking for pthread.h
-- Looking for pthread.h - found
-- Performing Test CMAKE_HAVE_LIBC_PTHREAD
-- Performing Test CMAKE_HAVE_LIBC_PTHREAD - Failed
-- Looking for pthread_create in pthreads
-- Looking for pthread_create in pthreads - not found
-- Looking for pthread_create in pthread
-- Looking for pthread_create in pthread - found
-- Found Threads: TRUE
-- Configuring done
-- Generating done
-- Build files have been written to: /mnt/d/unsa/is/lab4/cpp copy/build
villa7523@LAPTOP-E0T5UEV5:/mnt/d/unsa/is/lab4/cpp copy/build$ make
```



```
villa7523@LAPTOP-E0T5UEV5:/mnt/d/unsa/is/lab4/cpp copy/build$ make
[ 20%] Built target gtest
Scanning dependencies of target sample1Unittest
[ 30%] Building CXX object CMakeFiles/sample1Unittest.dir/sample1Unittest.c
c.o
[ 40%] Linking CXX executable sample1Unittest
[ 40%] Built target sample1Unittest
Scanning dependencies of target gmock
[ 50%] Building CXX object googletest/googlemock/CMakeFiles/gmock.dir/src/g
mock-all.cc.o
[ 60%] Linking CXX static library ../../lib/libgmock.a
[ 60%] Built target gmock
Scanning dependencies of target gmock_main
[ 70%] Building CXX object googletest/googlemock/CMakeFiles/gmock_main.dir/
src/gmock_main.cc.o
[ 80%] Linking CXX static library ../../lib/libgmock_main.a
[ 80%] Built target gmock_main
Scanning dependencies of target gtest_main
[ 90%] Building CXX object googletest/googletest/CMakeFiles/gtest_main.dir/
src/gtest_main.cc.o
[100%] Linking CXX static library ../../lib/libgtest_main.a
[100%] Built target gtest_main
```



```
villa7523@LAPTOP-E0T5UEV5:/mnt/d/unsa/is/lab4/cpp copy/build$
villa7523@LAPTOP-E0T5UEV5:/mnt/d/unsa/is/lab4/cpp copy/build$
[ RUN      ] FactorialTest.Zero
[       OK ] FactorialTest.Zero (0 ms)
[ RUN      ] FactorialTest.Positive
[       OK ] FactorialTest.Positive (0 ms)
[----------] 3 tests from FactorialTest (0 ms total)

[----------] 3 tests from IsPrimeTest
[ RUN      ] IsPrimeTest.Negative
[       OK ] IsPrimeTest.Negative (0 ms)
[ RUN      ] IsPrimeTest.Trivial
[       OK ] IsPrimeTest.Trivial (0 ms)
[ RUN      ] IsPrimeTest.Positive
[       OK ] IsPrimeTest.Positive (0 ms)
[----------] 3 tests from IsPrimeTest (0 ms total)

[----------] Global test environment tear-down
[==========] 6 tests from 2 test suites ran. (0 ms total)
[  PASSED  ] 6 tests.
villa7523@LAPTOP-E0T5UEV5:/mnt/d/unsa/is/lab4/cpp copy/build$
```