Análisis de coberbura usando **gcov** y **lcov**

Departamento de Computación, FCEyN, Universidad de Buenos Aires.

1er Cuatrimestre 2020

Introducción

- Esta guía introduce dos herramientas para el análisis de cobertura de un programa.
- Las dos instrucciones son GCOV y LCOV.
- Gracias a ellas, se puede generar automáticamente un reporte gráfico de la cobertura.
- Nosotros nos basaremos en Test Suites generados para llegar a un 100 % de cobertura.
- Análisis de cobertura con CLION.

GCOV

El GCOV ya viene incluido en el compilador **gcc** del CLION. Esta herramienta genera archivos adicionales durante la ejecución, incluyendo en el CMakeList las siguientes lineas:

```
# -g: Hace que podamos debuggear el programa.
# --coverage: Hace que se pueda calcular el cubrimiento de los casos de test.
set(CMAKE_CXX_FLAGS "-g --coverage")
# Un temita de gcov que no se lleva tan bien con CMake,
# y entonces hay que poner esta línea.
# Para más información, ver https://texus.me/2015/09/06/cmake-and-gcov/.
set(CMAKE_CXX_OUTPUT_EXTENSION_REPLACE 1)
```

Suponiendo que nuestro programa principal se denomina **main.cpp** durante la compilación son generados dos archivos adicionales: **main.gcno** y **main.gcda**.

Se pueden encontrar en el directorio de compilacion:

`cmake-build-debug/CMakeFiles/LaboTesting.dir/`.

LCOV

La aplicación lcov tiene que ser ejecutada desde línea de comando, una vez que fueron generados los .gcno y .gcda. Como ejemplo supondremos que, desde línea de comando vamos hasta el directorio conde están estos archivos.

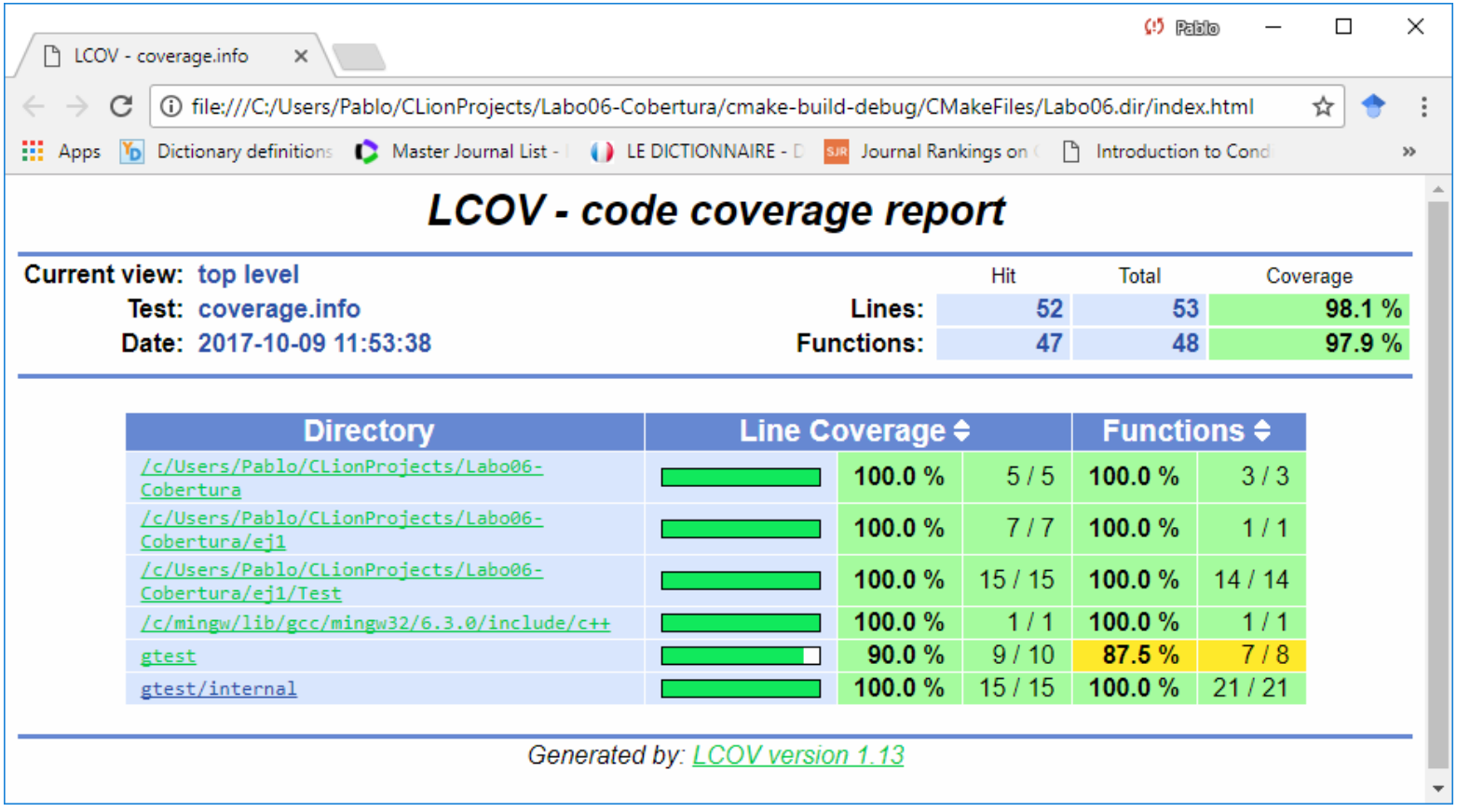
```
cmake-build-debug/CMakeFiles/LaboTesting.dir> lcov --capture
      --directory . --output-file pnegri/labo07/cobertura/coverage.info
```

Esta instruccion de **lcov** crea en el *pnegri*/*labo*07/*cobertura*/ el archivo *coverage.info*.
El siguiente paso es generar el reporte en formato html con la instruccion **genhtml**. Para ello podemos ubicarnos en el directorio donde fue generado el archivo coverage.info, en nuestro caso *pnegri*/*labo*07/*cobertura*/:

```
pnegri/labo07/cobertura/> genhtml coverage.info --output-directory .
```

LCOV

En el directorio *pnegri/labo07/cobertura/*, se genero un reporte en html de la cobertura del programa. Para verlo, simplemente se abre el archivo **index.html** generado.



LCOV - Instalación

Linux

- El LCOV es un proyecto de git que se encuetra en `https://github.com/linux-test-project/lcov`. Para instalarlo podemos usar la guia o correr las siguientes instrucciones:
  - Chequer si el compilador ya está instalado ejecutando en la terminal: `g++ --version`.
  - Si está instalado verán versión del compilador.
  - Si no está instalado ejecutar: `sudo apt install g++`
  - Chequer si LCOV ya está instalado ejecutando en la terminal: `lcov --version`
  - Si está instalado verán versión de lcov.
  - Si no está instalado ejecutar: `sudo apt install lcov`

LCOV - Instalación

Windows

- Se debe crear un entorno unix para correr el LCOV.
  - El primer paso es instalar MSYS2[1]: adjuntamos un tutorial en el folder el TP.
  - Otro tutorial[2]. Para instalar paquetes en el MSYS2 se utiliza desde su consola:

    `pacman -S PAQUETE`

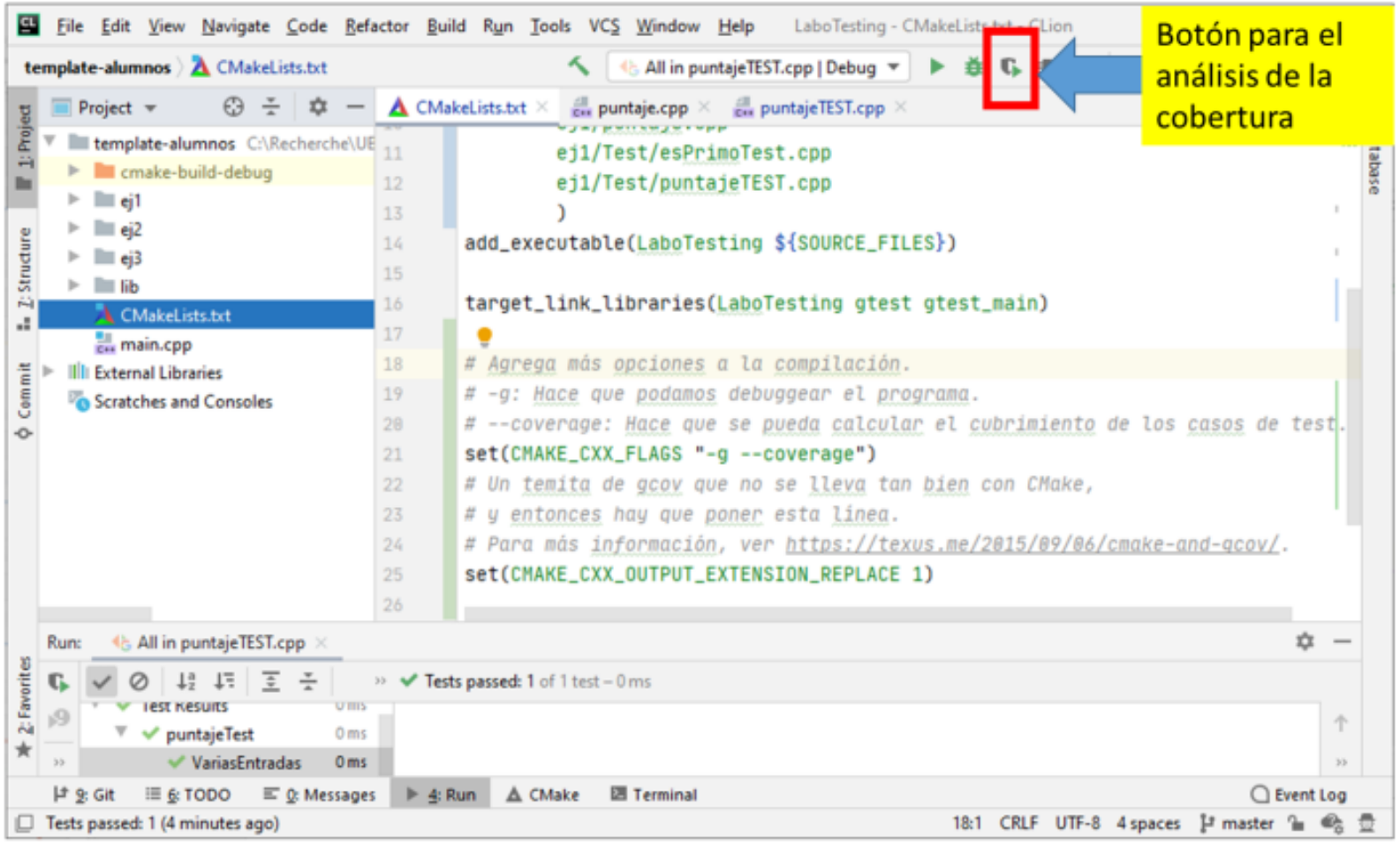
    Se deben instalar al menos los siguientes paquetes: git, gcc, make, binutils.

[1] https://www.msys2.org/

[2] https://txt.arboreus.com/2015/05/29/howto-get-lcov-test-coverage-on-windows.html

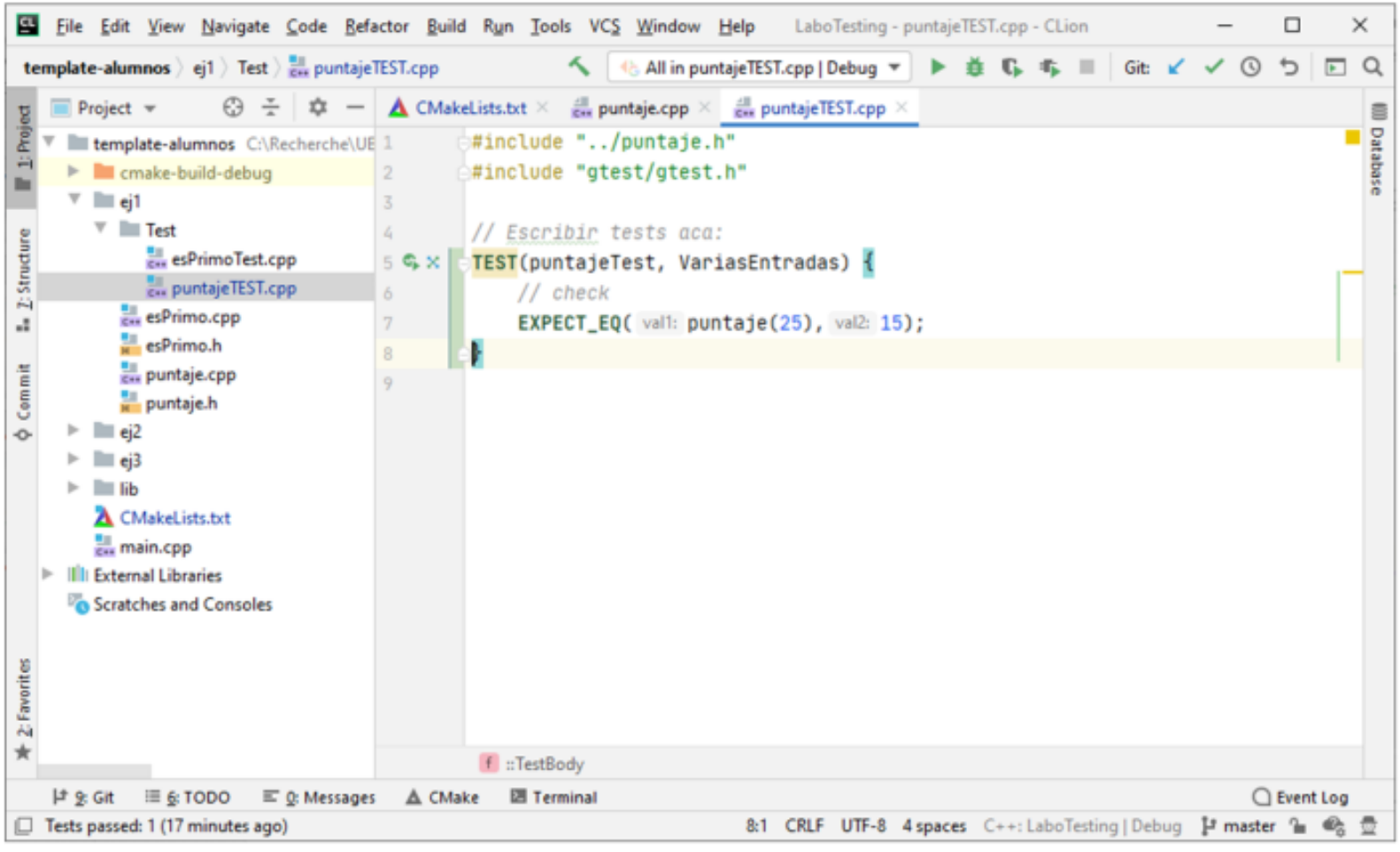
Análisis de cobertura con CLION

CLION, desde su versión 2019, posee un plugin que permite realizar un análsis de cobertura desde su entorno gráfico. Tomamos como ejemplo el laboratorio 07. Vemos también agregadas las líneas para generar la cobertura en el CMakeList.txt.

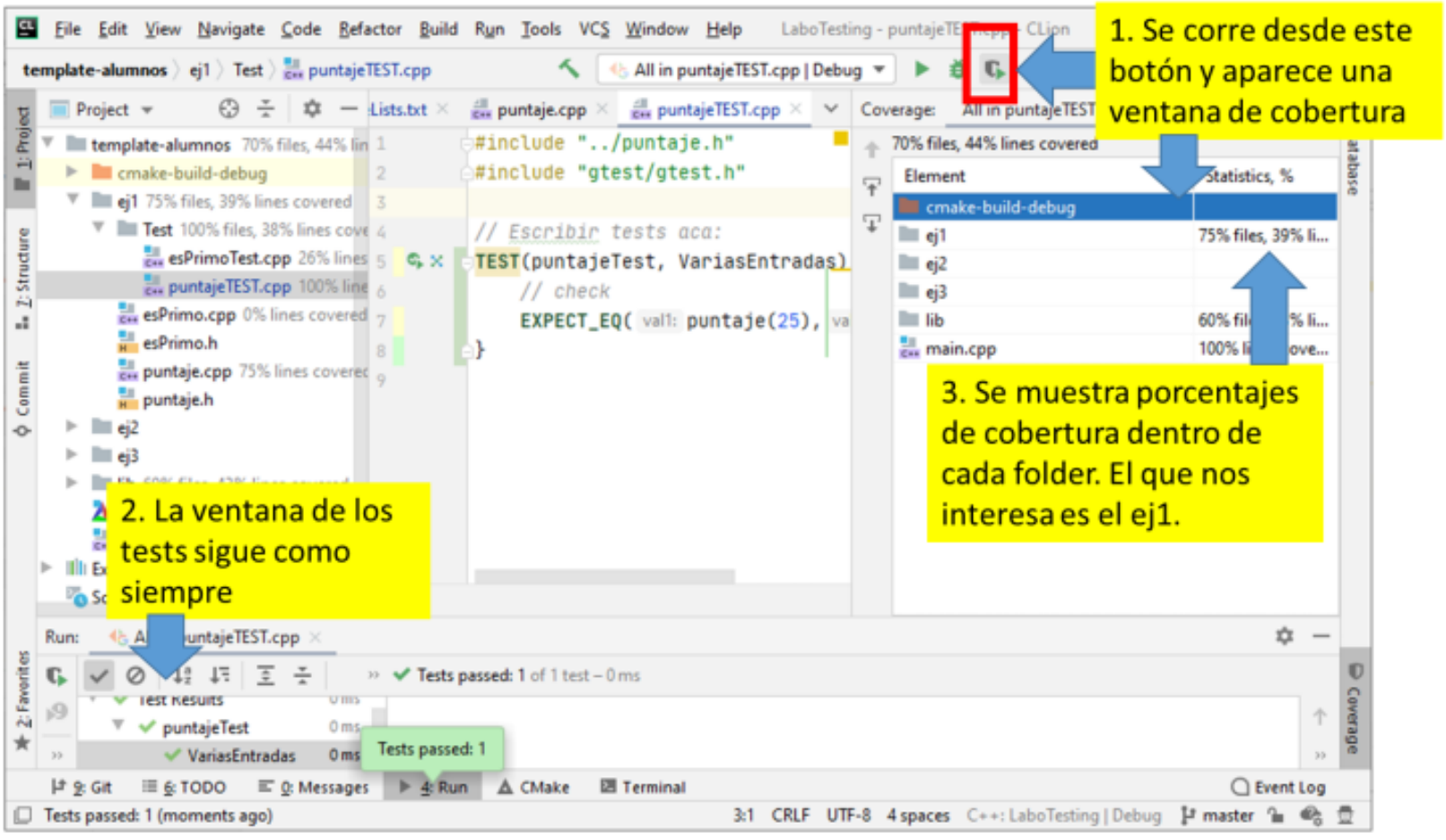


Análisis de cobertura con CLION

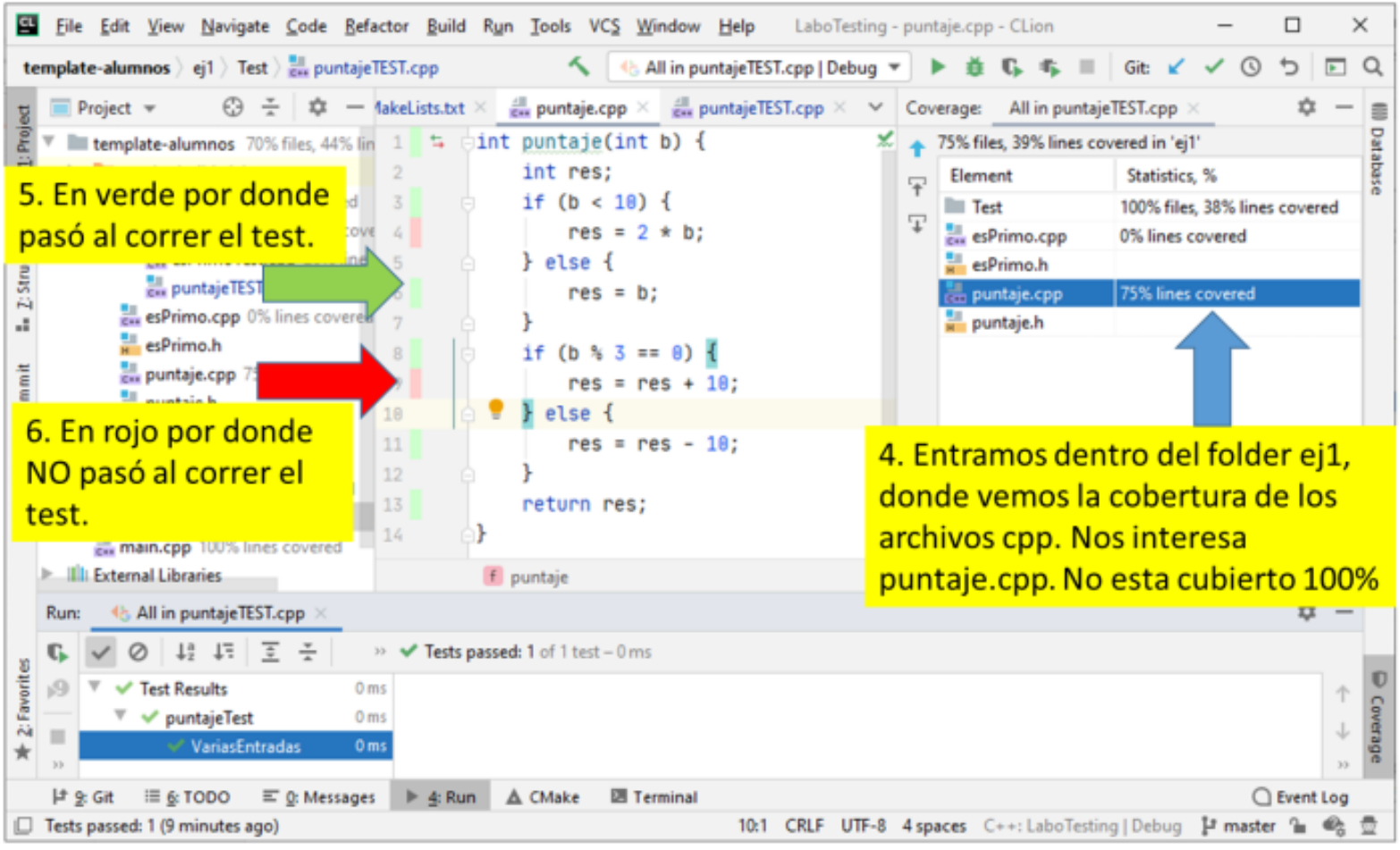
Tenemos generado un test para que se ejecute la funcion puntaje, en el archivo puntaje.cpp.



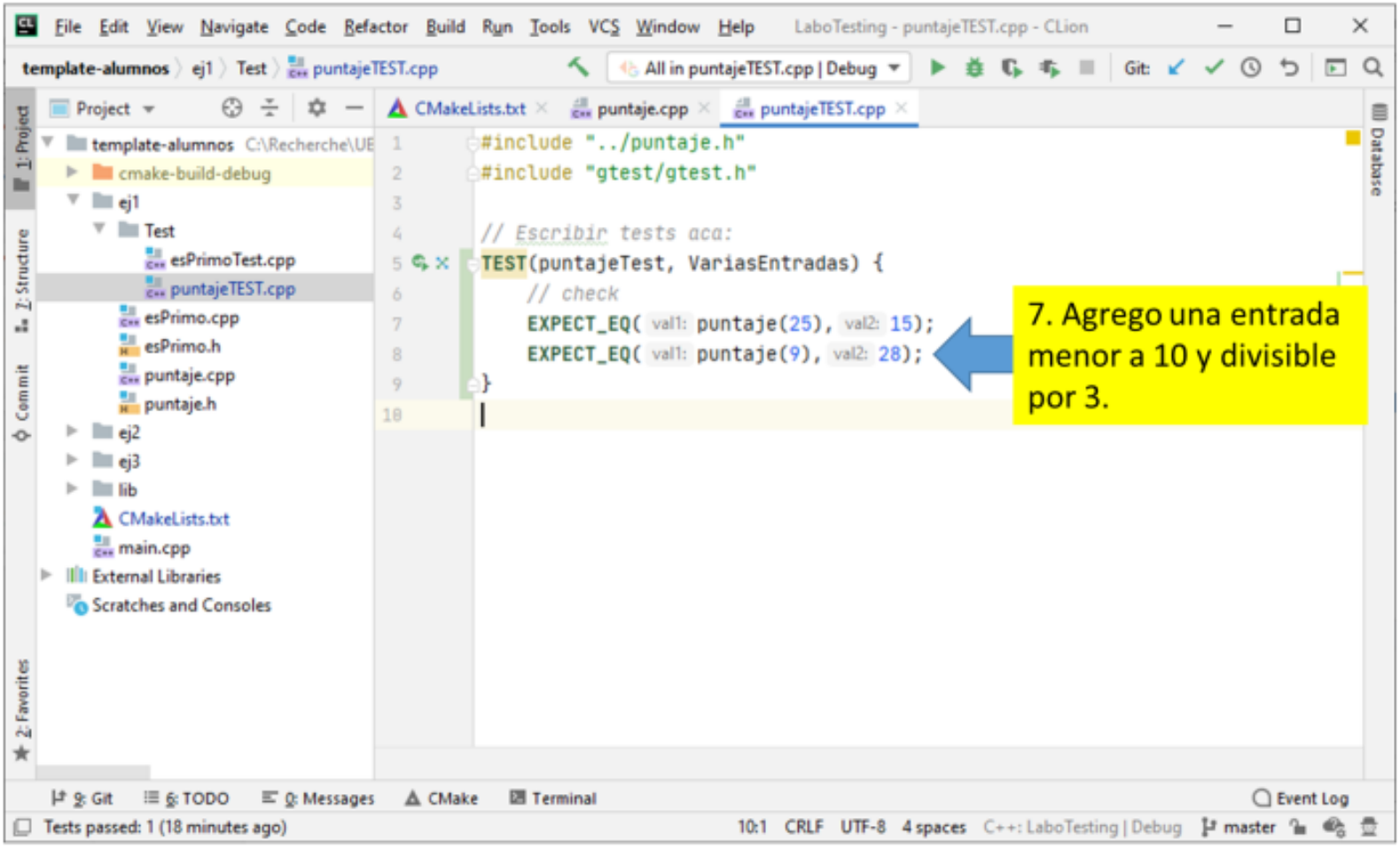
Análisis de cobertura con CLION



Análisis de cobertura con CLION



Análisis de cobertura con CLION



Análisis de cobertura con CLION