Quality

CppUTest

Unit testing and mocking framework for C/C++

- CppUTest Manual
- CppUMock Manual
- Plugin Manual
- GitHub Repo

Valgrind

Use *Valgrind* to test for memory leaks:[1]

```
valgrind --tool=memcheck --leak-check=full --track-origins=yes --show-reachable=yes --show-error-list=yes --error-limit=no <executable>
```

or, if this produces too much warnings about uninitialised values:

```
valgrind --tool=memcheck --leak-check=full --undef-value-errors=no --show-reachable=yes --show-error-list=yes --error-limit=no <executable>
```

CTest

- Kitware, Inc.: Testing With CTest
- Kitware, Inc.: Testing With CMake and CTest

Valgrind can also be used with *CTest*.[2]

```
ctest -V -C Debug -T memcheck
```

when putting this line into `CMakeLists.txt`:

```
set(MEMORYCHECK_SUPPRESSIONS_FILE
"${CMAKE_SOURCE_DIR}/.valgrind_suppressions" CACHE FILEPATH "" FORCE)
include(CTest)
```

Misc

- Google: AddressSanitizer (*ASan*): compile/link with flag `-fsanitize=address`
- Google: *Breakpad*
- Gimpel Software: PC-Lint

[1] See also the command line options page of Valgrind Memchk for details.

[2] For details see the CTest MemCheck Step documentation.

From:
https://wiki.rho62.de/ - **rho62 Wiki**

Permanent link:
https://wiki.rho62.de/doku.php?id=development:tools:quality

Last update: **2023/08/25 08:27**