

TE000 Test Report on the Library codecs_lib

Conventions

Each test is defined following the same format. Each test receives a unique test identifier and a reference to the ID(s) of the requirements it covers (if applicable). The goal of the test is described to clarify what is to be tested. The test steps are described in brief but clear instructions. For each test it is defined what the expected results are for the test to pass. Finally, the test result is given, this can be only pass or fail.

The test format is as follows:

Test Identifier: TEST-[I/A/D/T]-XYZ

Requirement ID(s): REQ-uvw-xyz

Verification method: I/A/D/T

Test goal: Description of what is to be tested

Expected result: What test result is expected for the test to pass

Test steps: Step by step instructions on how to perform the test

Test result: PASS/FAIL

The test ID starts with the fixed prefix 'TEST'. The prefix is followed by a single letter, which defines the test type / verification method. The last part of the ID is a 3-digits *hexadecimal* number (0..9|A..F), with the first digit identifying the module, the second digit identifying a class / function, and the last digit - the test ordering number for this object. E.g. 'TEST-T-112'. Each test type has its own counter, thus 'TEST-T-112' and 'TEST-A-112' tests are different entities, but they refer to the same object (class or function) within the same module.

The verification method for a requirement is given by a single letter according to the table below:

Term	Definition
Inspection (I)	Control or visual verification
Analysis (A)	Verification based upon analytical evidences
Test (T)	Verification of quantitative characteristics with quantitative measurement
Demonstration (D)	Verification of operational characteristics without quantitative measurement

Tests definition (Inspection)

Test Identifier: TEST-I-000

Requirement ID(s): REQ-INT-000

Verification method: I

Test goal: Reliable dependencies.

Expected result: If any 3rd party library is used, it should be widely accepted, well maintained and not marked for deprecation in the nearest future (2 years).

Test steps: Check / review the source code.

Test result: PASS

Test Identifier: TEST-I-001

Requirement ID(s): REQ-UDR-000

Verification method: I

Test goal: Library's documentation.

Expected result: The requirements documentation, test reports and the user / programmer reference documentation is present. Reference documents provide 'use manuals' sufficient to work with the library without software development experience (via frontend), and the implementation is also documented sufficiently (API reference) to ensure the maintenance and future modification of the library.

Test steps: Check / review the documentation provided with the library.

Test result: PASS

Tests definition (Analysis)

Test Identifier: TEST-A-000

Requirement ID(s): REQ-FUN-000

Verification method: A

Test goal: The library implements the COBS encoding / decoding properly.

Expected result: The functionality is implemented and properly tested with respect to the requirements defined in the [RE001](#) document.

Test steps: Check / review the source code. Run the unit-test suits defined in the [ut001_cobs.py](#) module. Report results in the [TE001](#) document.

Test result: PASS

Test Identifier: TEST-A-001

Requirement ID(s): REQ-FUN-001

Verification method: A

Test goal: The library implements the Vigenere cipher encoding / decoding properly.

Expected result: The functionality is implemented and properly tested with respect to the requirements defined in the [RE002](#) document.

Test steps: Check / review the source code. Run the unit-test suits defined in the [ut002_vigenere.py](#) module. Report results in the [TE002](#) document.

Test result: PASS

Test Identifier: TEST-A-002

Requirement ID(s): REQ-FUN-002

Verification method: A

Test goal: The library implements the WH random generator encoding / decoding properly.

Expected result: The functionality is implemented and properly tested with respect to the requirements defined in the [RE003](#) document.

Test steps: Check / review the source code. Run the unit-test suits defined in the [ut003_wichmann_hill.py](#) module. Report results in the [TE003](#) document.

Test result: PASS

Test Identifier: TEST-A-003

Requirement ID(s): REQ-FUN-003

Verification method: A

Test goal: The library implements a simple, key-less data scrambler / un-scrambler.

Expected result: The functionality is implemented and properly tested with respect to the requirements defined in the [RE004](#) document.

Test steps: Check / review the source code. Run the unit-test suits defined in the [ut004_xor_scrambler.py](#) module. Report results in the [TE004](#) document.

Test result: PASS

Tests definition (Demonstration)

Test Identifier: TEST-D-000

Requirement ID(s): REQ-IAR-000, REQ-IAR-002

Verification method: D

Test goal: System requirements check.

Expected result: The dependencies check script (module) is present and it detects the missing dependencies, too old versions as well as improper or too old Python interpreter version. If this script does

not detect problems, the library should operate as designed.

Test steps: Install the library on different PCs using different OS. Run the check module [check_dependencies.py](#). If missing dependencies or improper Python version is detected take the corrective actions. When the dependencies check has passed - try to work with the library. See also [tested_OS](#).

Test result: PASS

Test Identifier: TEST-D-001

Requirement ID(s): REQ-IAR-001

Verification method: D

Test goal: Multi-platform functionality.

Expected result: The library works as designed under MS Windows and GNU Linux OS, at least.

Test steps: Install the library and try to work with it on different PCs using different OS. Mark the tested OS + Python version in the [tested_OS.md](#) document as well as the result of the test.

Test result: PASS

Traceability

For traceability the relation between tests and requirements is summarized in the table below:

Requirement ID	Covered in test(s)	Verified [YES/NO])
REQ-FUN-000	TEST-A-000	YES
REQ-FUN-001	TEST-A-001	YES
REQ-FUN-002	TEST-A-002	YES
REQ-FUN-003	TEST-A-003	YES
REQ-INT-000	TEST-I-000	YES
REQ-IAR-000	TEST-D-000	YES
REQ-IAR-001	TEST-D-001	YES
REQ-IAR-002	TEST-D-000	YES
REQ-UDR-000	TEST-I-001	YES
Software ready for production [YES/NO]		Rationale
YES		All tests are passed