

```

UsuarioT@DESKTOP-0AK4NBH MINGW64 ~
$ cd Documents/ReposGit/dawb1-2425-ejercicios-u1-Trevictus/

UsuarioT@DESKTOP-0AK4NBH MINGW64 ~/Documents/ReposGit/dawb1-2425-ejercicios-u1-Trevictus (main)
$ pytest
===== test session starts =====
platform win32 -- Python 3.12.6, pytest-8.3.3, pluggy-1.5.0
rootdir: C:\Users\UsuarioT\Documents\ReposGit\dawb1-2425-ejercicios-u1-Trevictus
collected 8 items

tests\test_prueba1.py ..... [ 62%]
tests\test_suma.py ... [100%]

===== 8 passed in 0.03s =====

UsuarioT@DESKTOP-0AK4NBH MINGW64 ~/Documents/ReposGit/dawb1-2425-ejercicios-u1-Trevictus (main)
$ |

```

```

platform win32 -- Python 3.12.6, pytest-8.3.3, pluggy-1.5.0
rootdir: C:\Users\UsuarioT\Documents\ReposGit\dawb1-2425-ejercicios-u1-Trevictus
collected 8 items

tests\test_prueba1.py ..FFF [ 62%]
tests\test_suma.py ... [100%]

===== FAILURES =====
_____ test_retornar_mayor[5-8-8] _____

num1 = 5, num2 = 8, expected = 8

    @pytest.mark.parametrize(
        "num1, num2, expected",
        [
            (0, 0, 0),
            (4, 4, 0),
            (5, 8, 8),
            (9, 3, 9),
            (-1, -3, -1)
        ]
    )
    def test_retornar_mayor(num1, num2, expected):
>     assert retornar_mayor(num1, num2) == expected
E       assert 0 == 8
E       + where 0 = retornar_mayor(5, 8)

tests\test_prueba1.py:15: AssertionError
_____ test_retornar_mayor[9-3-9] _____

num1 = 9, num2 = 3, expected = 9

    @pytest.mark.parametrize(
        "num1, num2, expected",
        [
            (0, 0, 0),
            (4, 4, 0),
            (5, 8, 8),
            (9, 3, 9),
            (-1, -3, -1)
        ]
    )
    def test_retornar_mayor(num1, num2, expected):
>     assert retornar_mayor(num1, num2) == expected
E       assert 0 == 9
E       + where 0 = retornar_mayor(9, 3)

tests\test_prueba1.py:15: AssertionError
_____ test_retornar_mayor[-1--3--1] _____

num1 = -1, num2 = -3, expected = -1

    @pytest.mark.parametrize(
        "num1, num2, expected",
        [
            (0, 0, 0),
            (4, 4, 0),
            (5, 8, 8),
            (9, 3, 9),
            (-1, -3, -1)
        ]
    )
    def test_retornar_mayor(num1, num2, expected):
>     assert retornar_mayor(num1, num2) == expected
E       assert 0 == -1
E       + where 0 = retornar_mayor(-1, -3)

tests\test_prueba1.py:15: AssertionError

===== short test summary info =====
FAILED tests/test_prueba1.py::test_retornar_mayor[5-8-8] - assert 0 == 8
FAILED tests/test_prueba1.py::test_retornar_mayor[9-3-9] - assert 0 == 9
FAILED tests/test_prueba1.py::test_retornar_mayor[-1--3--1] - assert 0 == -1

```