

P1.2e - Primeras pruebas unitarias - FAM

Sin error:

```
Simbolo del sistema
C:\Users\UsuarioT\Documents\ReposGit\prog1>pytest ./test
===== test session starts =====
platform win32 -- Python 3.12.6, pytest-8.3.3, pluggy-1.5.0
rootdir: C:\Users\UsuarioT\Documents\ReposGit\prog1
collected 9 items

test\test_prueba1.py ..... [ 66%]
test\test_suma.py ... [100%]

===== 9 passed in 0.03s =====
C:\Users\UsuarioT\Documents\ReposGit\prog1>
```

Con error:

```
Seleccionar Símbolo del sistema
C:\Users\UsuarioT\Documents\ReposGit\prog1>pytest ./test
===== test session starts =====
platform win32 -- Python 3.12.6, pytest-8.3.3, pluggy-1.5.0
rootdir: C:\Users\UsuarioT\Documents\ReposGit\prog1
collected 9 items

test\test_prueba1.py ..F... [ 66%]
test\test_suma.py ... [100%]

===== FAILURES =====
_____ test_comprobar_numero_params[-4f-patata-False] _____
input_x = '-4f', input_y = 'patata', expected = False

    @pytest.mark.parametrize(
        "input_x, input_y, expected",
        [
            ("1", "2", True),
            ("3", "-5", True),
            ("-4f", "patata", False)
        ]
    )
    def test_comprobar_numero_params(input_x, input_y, expected):
>         assert comprobar_numero(input_x, input_y) == expected
E       ValueError: invalid literal for int() with base 10: 'patata'

test\test_prueba1.py:22:
-----
numero1 = '-4f', numero2 = 'patata'

    def comprobar_numero(numero1, numero2):

        numero1 = numero1.strip()
        numero2 = int(numero2.strip())
E       ValueError: invalid literal for int() with base 10: 'patata'

src\prueba1.py:18: ValueError
===== short test summary info =====
FAILED test\test_prueba1.py::test_comprobar_numero_params[-4f-patata-False] - ValueError: invalid literal for int() with base 10: 'patata'
===== 1 failed, 8 passed in 0.13s =====
C:\Users\UsuarioT\Documents\ReposGit\prog1>
```