Coverage for **tests\unit\test_diffusion2d_functions.py**: 97%

34 statements | 33 run | 1 missing | 0 excluded

« prev ^ index » next coverage.py v7.6.10, created at 2025-01-17 16:38 +0100

```
1  """
2  Tests for functions in class SolveDiffusion2D
3  """
4
5  import unittest
6  from unittest import TestCase
7  from diffusion2d import SolveDiffusion2D
8  import numpy as np
9
10 class TestDiffusion2D(TestCase):
11
12     def setUp(self):
13         self.solver = SolveDiffusion2D()
14
15     def test_initialize_domain(self):
16         """
17         Check function SolveDiffusion2D.initialize_domain
18         """
19         w, h, dx, dy = 2.0, 3.0, 0.5, 0.5
20         expected_nx = 4  # int(2.0 / 0.5)
21         expected_ny = 6  # int(3.0 / 0.5)
22
23         self.solver.initialize_domain(w, h, dx, dy)
24
25         self.assertEqual(self.solver.nx, expected_nx)
26         self.assertEqual(self.solver.ny, expected_ny)
27
28     def test_initialize_physical_parameters(self):
29         """
30         Checks function SolveDiffusion2D.initialize_physical_parameters
31         """
32         d, T_cold, T_hot = 1.0, 250.0, 300.0
33         self.solver.dx = 0.5
34         self.solver.dy = 0.25
35         expected_dt = 0.025  # (0.5**2 * 0.25**2) / (2 * 1.0 * (0.5**2 + 0.25**2))
36
37         self.solver.initialize_physical_parameters(d, T_cold, T_hot)
38
39         self.assertEqual(self.solver.D, d)
40         self.assertEqual(self.solver.T_cold, T_cold)
41         self.assertEqual(self.solver.T_hot, T_hot)
42         self.assertEqual(self.solver.dt, expected_dt)
43
44     def test_set_initial_condition(self):
45         """
46         Checks function SolveDiffusion2D.set_initial_condition
```

```
47          """
48          self.solver.nx, self.solver.ny = 4, 4
49          self.solver.dx, self.solver.dy = 1.0, 1.0
50          self.solver.T_cold = 25.0
51          self.solver.T_hot = 30.0
52          expected_u = np.array([
53              [25.0, 25.0, 25.0, 25.0],
54              [25.0, 25.0, 25.0, 25.0],
55              [25.0, 25.0, 25.0, 25.0],
56              [25.0, 25.0, 25.0, 25.0]
57          ])
58
59          u = self.solver.set_initial_condition()
60
61          np.testing.assert_array_equal(u, expected_u)
62
63  if __name__ == '__main__':
64      unittest.main()
```