- ◯ TT5L (Thursday)

5. What is the output of add(5, 5)? (1 Point) *

10

6. What is returned by subtract(10, 15)? (1 Point) *

-5

7. What does multiply(3, -3) return? (1 Point) *

-9

8. The function divide() always returns an integer. (1 Point) *

- ◯ True
- ◉ False

9. What is the correct output of modulus(14, 4)? (1 Point) *

2

10. What exception is raised when you run divide(1, 0)? (1 Point) *

ValueError: Cannot divide by zero.

11. test_divide_by_zero() verifies exception handling (1 Point) *

- [x] True
- [ ] False

12. You notice that test_add() only tests positive and negative values. You want to test the **identity property** of addition. Which of the following test cases best represents this? (1 Point) *

- [ ] self.assertEqual(add(1, 1), 2)
- [ ] self.assertEqual(add(5, 5), 10)
- [x] self.assertEqual(add(0, 99), 99)
- [ ] self.assertEqual(add(-1, -1), -2)

13. To test **equivalence partitioning** in divide(), which test checks behavior when a negative divisor is used? (1 Point) *

- [x] self.assertEqual(divide(10, -2), -5.0)
- [ ] self.assertEqual(divide(-10, 2), 5.0)
- [ ] self.assertEqual(divide(0, 10), 0)
- [ ] self.assertEqual(divide(2, 2), 1)

14. All current modulus tests use even numbers as divisors. What result would you get from modulus(11, 3)? (1 Point) *

2

15. You're conducting **white-box testing** to ensure the if b == 0 block in divide() is covered. Which test achieves this? (1 Point) *

- [ ] self.assertEqual(divide(0, 5), 0)
- [ ] self.assertRaises(ValueError, divide(5, 0))
- [x] with self.assertRaises(ValueError): divide(5, 0)

- ◯ self.assertEqual(divide(5, 1), 5)

16. The following test method is added, but something is wrong:

```
def test_divide_typo(self):
    divide(10, 2)
```

What is the **problem** with this test?
(1 Point) *

- ◯ $divide()functionisundefined$
- ◯ $Thereisamissingimport$
- ◉ $Noassertionismadetocheckcorrectness$
- ◯ $Thefunctioncallisinvalid$

17. You are asked to verify that the subtract() function **never returns a string**. Write a test to check the return type of subtract(10, 2). What assertion would pass? (1 Point) *

- ◉ self.assertTrue(isinstance(subtract(10, 2), int))
- ◯ self.assertFalse(type(subtract(10, 2)) == int)
- ◯ self.assertEqual(subtract(10, 2), "8")

- ◯ self.assertRaises(TypeError, subtract(10, 2))

18. You add the following test:

    self.assertEqual(modulus(5, 10), 5)

    What is the actual output when this test is run? (1 Point) *

- ◯ It fails, returns 0
- ◉ It passes ✅
- ◯ It raises a ZeroDivisionError
- ◯ It returns None

19. You're told one of the tests is written to **intentionally fail**. Which test below would definitely fail? (1 Point) *

- ◯ self.assertEqual(add(1, 1), 2)
- ◯ self.assertEqual(divide(4, 2), 2)
- ◉ self.assertEqual(modulus(10, 3), 2)
- ◯ self.assertEqual(subtract(5, 2), 3)

20. A developer accidentally changes this line:

```
def multiply(a, b):
    return a * b + 1
```

Which test will **fail** after this change?
(1 Point) *

- ○ self.assertEqual(multiply(3, 3), 9)
- ○ self.assertEqual(multiply(0, 3), 0)
- ○ self.assertEqual(multiply(5, 5), 25)
- ◉ All of the above

21. Upload your Script here (test_calculator.py) -> SAVE IT AS PDF (1 Point) (Non-anonymous question) *

test_calculator.pdf

File number limit: 1 Single file size limit: 100MB
Allowed file types: Word, Excel, PPT, PDF, Image, Video, Audio