

# 华中科技大学

# 课程实验报告

课程名称： 软件工程设计实验

专业班级： 软件 2203 班

学 号： U202217238

姓 名： 刘杰

指导教师： 刘小峰

报告日期： 2024 年 4 月 8 日

软件学院

## 目 录

<b>1 TDD 测试驱动开发实验 .....</b>	<b>1</b>
1.1 实验目的 .....	<b>1</b>
1.2 实验内容及要求 .....	<b>1</b>
1.3 实验内容报告 .....	1
1.4 实验小结 .....	4

# 1 TDD 测试驱动开发实验

## 1.1 实验目的

- (1) 理解 TDD 测试驱动开发的思想
- (2) 掌握 TDD 测试驱动开发的基本过程

## 1.2 实验内容及要求

- (1) 阅读：Kent Beck 《测试驱动开发》
- (2) 基于 TDD 开发一个判断字符串是 IP4 地址的功能
- (3) 使用任何 OO 语言
- (4) 使用 xUnit 测试框架

## 1.3 实验内容报告（以下代码基于 python 语言，测试框架为 pyUnit 测试框架）

### (1) 测试代码

```
import unittest
from TDD import IsIP4

class TDDTestCase(unittest.TestCase):
    def test_NegativeCharacter_IP4(self):
        self.assertEqual(IsIP4('abc.def.ghi.jkl'), False)

    def test_CorrectIP4_IP4(self):
        self.assertEqual(IsIP4('255.255.255.255'), True)

    def test_NotString_IP4(self):
        self.assertEqual(IsIP4(1), False)

    def test_InCorrectLengthOfIP4_IP4(self):
        self.assertEqual(IsIP4('1.2.3'), False)

    def test_InCorrectLocalLength_IP4(self):
```

```

        self.assertEqual(IsIP4('2555.12.1.2'), False)

    def test_InCorrectDelimiter_IP4(self):
        self.assertEqual(IsIP4('23/23/1/3'), False)

    def test_emptyString_IP4(self):
        self.assertEqual(IsIP4(''), False)

    def test_DifferentZeroIP4_IP4(self):
        self.assertEqual(IsIP4('0.0.00.000'), True)

    def test_FalseNumberDotIP4_IP4(self):
        self.assertEqual(IsIP4('1.1.1.1.1'), False)

    def test_ContinuousDot_IP4(self):
        self.assertEqual(IsIP4('255..255.255'), False)

if __name__ == '__main__':
    unittest.main()

```

## 具体测试案例解释

### (1) test\_NegativeCharacter\_IP4 测试案例

这个测试案例是为了验证对应函数能否正确将 IP4 中含有字母的情况判断为错误的 IP4 地址。

### (2) test\_CorrectIP4\_IP4 测试案例

这个测试案例是为了验证对应函数能否将正确的 IP4 地址判断为正确

### (3) test\_NotString\_IP4 测试案例

这个测试案例是为了验证对应函数能否将用户输入的内容不为字符串的情况判断为错误的 IP4 地址

### (4) test\_InCorrectLengthOfIP4\_IP4 测试案例

这个测试案例是为了验证对应函数能否将用户输入字符串低于正常 IP4 地址长度范围时的情况判断为错误的 IP4 地址

### (5) test\_InCorrectLocalLength\_IP4

这个测试案例是为了验证对应函数能否将输入字符串中内部有一段数字超出 255 的情况判断为错误的 IP4 地址

(6) test\_InCorrectDelimiter\_IP4

这个测试案例是为了验证对应函数能否将不同数字串之间不是以'.'为分隔符的情况判断为错误的 IP4 地址

(7) test\_emptyString\_IP4

这个测试案例是为了验证对应函数能否将用户输入字符串为空的情况判断为错误的 IP4 地址

(8) test\_DifferentZeroIP4\_IP4

这个测试用例是为了验证对应函数能否在用户以不同方式输入 0 的情况下输出正确的答案

(9) test\_FalseNumberDotIP4\_IP4

这个测试用例是为了验证对应函数能否将输入字符串中有过多或过少的'.'分隔符的情况判断为错误的 IP4 地址

(10) test\_ContinuousDot\_IP4

这个测试案例是为了验证对应函数能否将处理输入字符串中有连续点出现的情况判断为错误的 IP4 地址

## (2) 功能实现代码

```
def IsIP4(IP):
    if type(IP) == str:
        if len(IP) in range(7):
            return False
        else:
            sIP = []
            CorrectCharacter = '.0123456789'
            start = 0
            last = 0
            count = 0
            countDot = 0
            dotFlag = False
            for character in IP:
                count += 1
                if character not in CorrectCharacter:
                    return False
                elif character == '.':
                    dotFlag = True
```

```

        countDot += 1
    if dotFlag:
        if start == last:
            return False
        sIP.append(IP[start:last])
        start = last + 1
        last = start
        dotFlag = False
    elif count == len(IP):
        sIP.append(IP[start:last + 1])
    else:
        last += 1
    if countDot != 3:
        return False
    iIP1 = int(sIP[0])
    iIP2 = int(sIP[1])
    iIP3 = int(sIP[2])
    iIP4 = int(sIP[3])
    print(iIP1, iIP2, iIP3, iIP4)
    if iIP1 in range(256) and iIP2 in range(256) and iIP3
in range(256) and iIP4 in range(256):
        return True
    else:
        return False
else:
    return False

```

## 1.4 实验小结

1. 在本次实验的过程中，我理解了 TDD 测试驱动开发的思想，通过编写测试来驱动自己修改功能实现代码去通过对应的测试用例。这种思想可以让我及时发现代码的错误，提高代码质量和开发速度。
2. 通过本次实验，我掌握了 TDD 测试驱动开发的过程：先编写测试，再增添或修改代码，循环往复，直至对应功能可以将所有用户可能输入的测试案例形式输出正确的答案。