```python
import numpy as np
import numpy.random as npr
from test_util import *
from funkyyak import grad
npr.seed(1)

def test_getter():
    def fun(input_dict):
        A = np.sum(input_dict['item_1'])
        B = np.sum(input_dict['item_2'])
        C = np.sum(input_dict['item_2'])
        return A + B + C

    d_fun = grad(fun)
    input_dict = {'item_1' : npr.randn(5, 6),
                  'item_2' : npr.randn(4, 3),
                  'item_X' : npr.randn(2, 4)}

    result = d_fun(input_dict)
    assert np.allclose(result['item_1'], np.ones((5, 6)))
    assert np.allclose(result['item_2'], 2 * np.ones((4, 3)))
    assert np.allclose(result['item_X'], np.zeros((2, 4)))

def test_grads():
    def fun(input_dict):
        A = np.sum(np.sin(input_dict['item_1']))
        B = np.sum(np.cos(input_dict['item_2']))
        return A + B

    def d_fun(input_dict):
        g = grad(fun)(input_dict)
        A = np.sum(g['item_1'])
        B = np.sum(np.sin(g['item_1']))
```

```
34            C = np.sum(np.sin(g['item_2']))
35            return A + B + C
36
37        input_dict = {'item_1' : npr.randn(5, 6),
38                      'item_2' : npr.randn(4, 3),
39                      'item_X' : npr.randn(2, 4)}
40
41        check_grads(fun, input_dict)
42        check_grads(d_fun, input_dict)
```