

Our Solution(s)

Run Code

Solution 1

```
1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
2
3 class Program {
4     // O(nd) time | O(n) space
5     func minimumNumberOfCoinsForChange(target: Int, denominations: List<Int>) -> Int {
6         var numberOfCoins = Array<Int>(repeating: Int(Int16.max), count: target + 1)
7         numberOfCoins[0] = 0
8
9         for denomination in denominations {
10             for amount in 0 ..< numberOfCoins.count {
11                 if denomination <= amount {
12                     numberOfCoins[amount] = min(numberOfCoins[amount], numberOfCoins[amount - denomination] + 1)
13                 }
14             }
15         }
16
17         return numberOfCoins[target] != Int(Int16.max) ? numberOfCoins[target] : -1
18     }
19 }
20
```

Our Tests

Your Solutions

Run Code

Solution 1

Solution 2

Solution 3

```
1 class Program {
2     func minimumNumberOfCoinsForChange(target: Int, denominations: List<Int>) -> Int {
3         // Write your code here.
4         return -1
5     }
6 }
7
```

Custom Output

Submit Code

```

1  def test1():
2      @pytest.mark.parametrize('program', ['program1', 'program2'])
3      @pytest.mark.parametrize('test_case', ['test_case1', 'test_case2'])
4      def test(program, test_case):
5          # Test logic here
6          pass
7
8      # Test cases
9      test('program1', 'test_case1')
10     test('program1', 'test_case2')
11     test('program2', 'test_case1')
12     test('program2', 'test_case2')
13
14     # Run tests
15     pytest.main([__file__, '-v'])
16
17     # Submit code
18     submit_code(program, test_case)
19
20     # Run tests
21     pytest.main([__file__, '-v'])
22
23     # Submit code
24     submit_code(program, test_case)
25
26     # Run tests
27     pytest.main([__file__, '-v'])
28
29     # Submit code
30     submit_code(program, test_case)

```

Run or submit code when you're ready.