"Test Cases: Assignment 2"
"Problem 1"
"Testing: (set-equal? '(6 7 83 2) '(83 7 6 2)); Expected output: #t; Actual output:"
(set-equal? '(6 7 83 2) '(83 7 6 2))
"Testing: (set-equal? '(6 7 83) '(83 7 6 2)); Expected output: #f; Actual output:"
(set-equal? '(6 7 83) '(83 7 6 2))
"Problem 2"
"Testing: (RemoveFirst '3 '(8 833 (37) ((340)) 9 39)); Expected output: '(((340)) 9 39); Actual output:"
(RemoveFirst '3 '(8 833 (37) ((340)) 9 39))
"Testing: (RemoveFirst '10 '(8 833 (37) ((340)) 9 39)); Expected output: '(); Actual output:"
(RemoveFirst '10 '(8 833 (37) ((340)) 9 39))
"Testing: (RemoveFirst '-3 '(8 833 (37) ((340)) 9 39)); Expected output: #f; Actual output:"
(RemoveFirst '-3 '(8 833 (37) ((340)) 9 39))
"Problem 3"
"Testing: (RemoveLast '3 '(1 2 3 (4) ((5)) 6)); Expected output: '(1 2 3); Actual output:"
(RemoveLast '3 '(1 2 3 (4) ((5)) 6))
"Testing: (RemoveLast '10 '(1 2 3 (4) ((5)) 6)); Expected output: '(); Actual output:"
(RemoveLast '10 '(1 2 3 (4) ((5)) 6))
"Testing: (RemoveLast '-3 '(1 2 3 (4) ((5)) 6)); Expected output: #f; Actual output:"
(RemoveLast '-3 '(1 2 3 (4) ((5)) 6))
"Problem 4"
"Testing: (SliceList '1 '4 '(the who is the best band ever)); Expected output: (the who is the); Actual output:"
(SliceList '1 '4 '(the who is the best band ever))
"Testing: (SliceList '6 '4 '(the who is the best band ever)); Expected output: #f; Actual output:"
(SliceList '6 '4 '(the who is the best band ever))
"Testing: (SliceList '-3 '4 '(the who is the best band ever)); Expected output: #f; Actual output:"
(SliceList '-3 '4 '(the who is the best band ever))
"Testing: (SliceList '1 '20 '(the who is the best band ever)); Expected output: #f; Actual output:"
(SliceList '1 '20 '(the who is the best band ever))
"Problem 5"
"Testing: (pair '(fee fi) '(fo fum)); Expected output: ((fee fo)(fi fum)); Actual output:"
(pair '(fee fi) '(fo fum))
"Testing: (pair '(fee fi fu) '(fo fum)); Expected output: #f; Actual output:"
(pair '(fee fi fu) '(fo fum))
"Problem 6"
"Testing: (pair2 '(fee fi fu) '(fo fum)); Expected output: ((fee fo)(fi fum)); Actual output:"
(pair2 '(fee fi fu) '(fo fum))
"Testing: (pair2 '(fee fi) '(fo fum fu)); Expected output: ((fee fo)(fi fum)); Actual output:"
(pair2 '(fee fi) '(fo fum fu))
"Problem 7"
"Testing: (div '13 '3); Expected output: (4 1); Actual output:"

```
(div '13 '3)
"Testing: (div '-13 '3); Expected output: #f; Actual output:"
(div '-13 '3)
"Testing: (div '13 '-3); Expected output: #f; Actual output:"
(div '13 '-3)
"Testing: (div '13 '0); Expected output: #f; Actual output:"
(div '13 '0)
```