

6.00 Handout, Lecture 7
(Not intended to make sense outside of lecture)

```
x = 0.0
numIters = 100000
for i in range(numIters):
    x += 0.1
print x #prints 10000.0, because print automatically rounds
print repr(x)
print 10.0*x == numIters

def close(x, y, epsilon = 0.00001):
    return abs(x - y) < epsilon

if close(10.0*x, numIters):
    print 'Good enough'

def isPal(x):
    """requires x to be a list
       returns True if the list is a palindrome; False otherwise"""
    assert type(x) == list
    temp = x
    temp.reverse
    if temp == x:
        return True
    else:
        return False

def silly(n):
    """requires: n is an int > 0
       Gets n inputs from user
       Prints 'Yes' if the inputs are a palindrome; 'No' otherwise"""
    assert type(n) == int and n > 0
    for i in range(n):
        result = []
        elem = raw_input('Enter something: ')
        result.append(elem)
    if isPal(result):
        print 'Is a palindrome'
    else:
        print 'Is not a palindrome'

def isPalTest():
    L = [1, 2]
    result = isPal(L)
    print 'Should print False:', result
    L = [1, 2, 1]
    result = isPal(L)
    print 'Should print True:', result
```

MIT OpenCourseWare
<http://ocw.mit.edu>

6.00SC Introduction to Computer Science and Programming
Spring 2011

For information about citing these materials or our Terms of Use, visit: <http://ocw.mit.edu/terms>.