Practice Problem 5.1

I wrote the following program, which is supposed to delete every other item on the list stored in the variable numbers. By looking at the output, it is clear that it doesn't work? Why? Can you fix my program?

```
# prob5.1.py
numbers = [0, 1, 2, 3, 4, 5, 6, 7]

print( numbers )

# Remove every other item on the list
i = 0
while i < len( numbers ):
    del numbers[i]
    i = i + 2

print( numbers )

# Printed output
# [0, 1, 2, 3, 4, 5, 6, 7]
# [1, 2, 4, 5, 7]
```

The loop proceeds from the beginning of the list to the end. As items are deleted, the list index of subsequent items reduces by 1. Because "i" is incremented by 2 each time through the loop, indexes for the items to delete change after each list item deletion.

```
# Solution 1: Change i to increment by 1,
#             to reflect changing list length.
i = 0
while i < len( numbers ):
    del numbers[i]
    i = i + 1

# Solution 2: Loop from the end of the list to the beginning,
#             to avoid the impact of the changing list length.
#             Account for odd/even nature of list length.
i = len( numbers ) - 2 + ( len(numbers) % 2 )
while i >= 0:
    del numbers[i]
    i = i - 2
```

Practice Problem 5.2

Rewrite the following program in a functional programming style using map().

```
L = [1, 2, 3, 4, 5]

def squareIt( x ):
    return x * x

squares = []
for e in L:
    squares.append( squareIt(e) )

print( squares )
```

```
L = [1, 2, 3, 4, 5]

def squareIt( x ):
    return x * x

print ( map( squareIt, L ) )
```

Practice Problem 5.3

Rewrite the program in 5.2 using list comprehensions.

```
L = [1, 2, 3, 4, 5]

def squareIt( x ):
    return x * x

print( [x*x for x in L] )
```