# Problem Wk.14.2.3: Aliasing Instances

This problem examines the problem of shared instances. We'll use the following simple class to illustrate.

```
class MyClass:
   def __init__(self, v):
       self.v = v
```

---

## Part 1: Try 1

Consider the following code:

```
def lotsOfClass(n, v):
   one = MyClass(v)
   result = []
   for i in range(n):
       result.append(one)
   return result

class10 = lotsOfClass(10, 'oh')

class10[0].v = 'no'
```

1. What is the value of `class10[0].v`:
2. What is the value of `class10[3].v`:

---

## Part 2: Try 2

Define a new version of lotsOfClass that has separate instances of the objects in each location of the list.

```
def lotsOfClass(n, v):
    pass
```

---

## Part 3: Try 3

Define another version of lotsOfClass that has separate instances of the objects in each location of the list. Use `util.makeVectorFill` (see Software Documentation) to accomplish the same thing.

```
def lotsOfClass(n, v):
    pass
```

6.01SC Introduction to Electrical Engineering and Computer Science

Spring 2011