Increment, Twice

```
def makeIncr(init):
    return PrimitiveSM(lambda s, i: i+1,
                       lambda s: s,
                       lambda : init)

class Incr:
    def __init__(self, initialValue):
        self.value = initialValue
    def step(self, input):
        self.value = input + 1
        return self.currentOutput()
    def currentOutput(self):
        return self.value
```

In lecture, there was an inconsistency, which led me to say the transition function of the PrimitiveSM was wrong. In fact, the error was elsewhere, and is corrected by the red text

Generic Superclass

```
class SM:
    def run(self, n = 10):
        result = [self.currentOutput()]
        for i in range(10):
            result.append(self.step())
        return result

class Incr (SM):
    def __init__(self, initialValue): …
    def step(self, input=None): …
    def currentOutput(self): …
```