CASE – SINGLETON

package myentitites

import case.lang.System

namespace EntitiesNameSpace {

@module

String->Object->Main

#public class Program

[public Program(String [] args)

[EntityPool Pool = EntityPool.getEntityPool]

assert(Pool) //asserts that Pool exists and has a value

Stream (n) String

Int MyInt = EntityPool.getStreamMemory() //retrieve mem from pool

Int GetInt = EntityPool.get(“MyInt”) //pointer to MyInt using pool get

//get pointer to CurrentLocationInList from the pool

Int ListStatus = n.get(“CurrentLocationInList”)

]

#end class

**A more simpler example of Stateful language**

When the number of apples increases, I want the number of oranges to decrease.

Public class Apples

Int apples = 0;

Public class Oranges

**Stateful** Apples .apples = Int Oranges = “Orange Lovers” **Tie** (equiv ++)

If the following syntax holds true, when apples changes (decreases or increases) , the variable in class Oranges, Int Oranges, tied by the tied keyword, is assigned (because of the equiv keyword is assinging to the variable to t he left, ,in this case Int Oranges) increases. So essentially, when the number of apples increases, the number of oranges increases (by ++).