CASE PROGRAM – ENUM

import case.lang.System

namespace HelloWorld {

//String->Object->Main is a type constructor

String->Object->Main

#public class Program

assigner MyLaundryCost = { Shirts = 50.0f, Hoodies = 10.00f, Socks = “Empty” }

@requires

[public Program(String [] args)

System.out.println(@texts:“Hello World”)

Print sizeOf(“Size of args structure” + sizeof(args))

//instantiating them is the same as class

stream ( n ) PieInventory //create a PieInventory variable on stream

MyPieInventory (n) PieInventory = new PieInventory<>

]

String->Object->Joke

#public union Joke

// A union is a special class type that can hold only one of its non-static [data //members](http://en.cppreference.com/w/cpp/language/data_members) at a time.

Stream (n) Integer int1

Stream(n) Integer int2

Stream (n) String str1

Stream (n) String str2

]

#end union

String->Object

#public synchronized class laundry

//upon entering this class the mutex is initiated

//until the end of the routine, until the routine is

//over then it releases

[test()]

[test2()]

#end class

**//Struct**

//Structs are by default public while class are by default package level

#struct PieInventory

Integer ChocolatePies

Integer Pumpkin Pies  
 String PieBusinessName

RevenueThisQuarter

[public void sellPies()]

[public void eatPies()]  
 [public void givePies()]

#end struct PieInventory

}