CASE PROGRAM – SYNCHRONIZED

import case.lang.System

namespace HelloWorld {

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

String->Object->Main

#public class Program

[public Program(String [] args)

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

Apple apple = new RottenApple()

RottenApple rotten = new Apple()

]

**//Synchronized**

//Used in the declaration of a method or code block to acquire the [mutex](http://en.wikipedia.org/wiki/Mutex) //lock for an object while the current [thread](http://en.wikipedia.org/wiki/Thread_(computer_science)) executes the code.[10] For static/ //methods, the object locked is the class's Class. Guarantees that at most one //thread at a time operating on the same object executes that code. The //mutex lock is automatically released when execution exits the //synchronized code. Fields, classes and interfaces cannot be declared as

//*synchronized*.

@synchronized

[public Test()]

#end class