**JVM Homework**

The first problem with memory leak is encountered in the main thread where an instance of PlanetaBoss is created and that object keeps a reference to a large linked list contained of PlanetaEmployee instances, where each PlanetaEmployee object contains new instance of String which also leads to memory leak.

The second memory leak problem is encountered in the PlanetaCache class where a HashMap named PLANETA\_CACHE is created and contains large numbers of key-value pairs, where an instance of the PlanetaSinger class is used as a key and an int as a value.

The third problem that leads to memory leak is the creation of a lot of PlanetaTiket instances, where every object holds a new String instance and uses the finalize method which blocks the garbage collector from collecting the instance, until the Finalizer thread queue don’t get to the instance and execute its finalize method. In this case that does not happen because the Finalizer thread is sleeping and never executes that method for the PlanetaTiket instances.