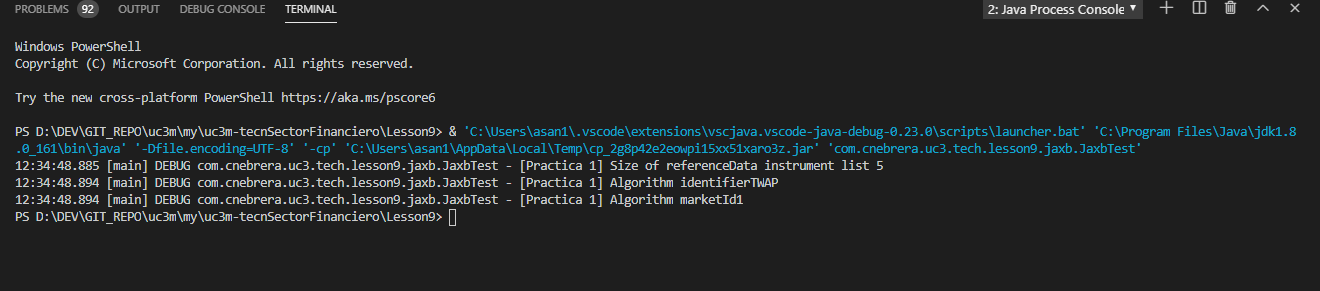
# Practica 9: Serialización de mensajes

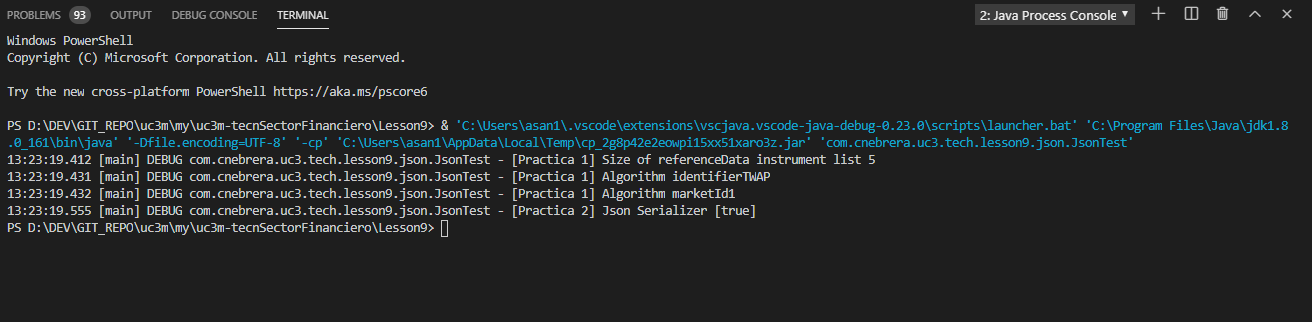
## Parte 1: Jaxb

A continuación, el resultado de correr la clase JaxbTest.java:



Parte 2: Json

A continuación, el resultado de correr la clase JsonTest.java:



Parte 5: Comparación de rendimiento

A continuación, para tomar los tiempos de rendimiento en manos se hizo con la JVM en caliente y se lanzaron 100 millones de veces las ejecuciones de serialización y deserialización por cada librería (jaxb, json, proto y kryo). Como resultado podemos ver a proto, la librería de Google como el ganador absoluto en pruebas de rendimiento, ya que es el que menos tarda en deserializar, serializar y el tamaño del objeto serializado es menor que los demás y con una diferencia importante correspondiente al sucesor en menor tamaño (Jaxb)

21:13:43.687 [main] DEBUG com.cnebrera.uc3.tech.lesson9.Measurement - [Practica 1] Size of referenceData instrument list 5

21:13:43.699 [main] DEBUG com.cnebrera.uc3.tech.lesson9.Measurement - [Practica 1] Algorithm identifierTWAP

21:13:43.700 [main] DEBUG com.cnebrera.uc3.tech.lesson9.Measurement - [Practica 1] Algorithm marketId1

21:13:43.787 [main] DEBUG com.cnebrera.uc3.tech.lesson9.Measurement - [Practica 2] Json Serializer [true]

21:13:43.848 [main] DEBUG com.cnebrera.uc3.tech.lesson9.Measurement - [Practica 3] Proto Serializer [true]

21:13:43.879 [main] DEBUG com.cnebrera.uc3.tech.lesson9.Measurement - [Practica 4] Kryo Serializer [true]

21:36:32.910 [main] DEBUG com.cnebrera.uc3.tech.lesson9.Measurement - [PerformanceSerialization] meanJaxb: 7502

21:36:32.910 [main] DEBUG com.cnebrera.uc3.tech.lesson9.Measurement - [PerformanceSerialization] meanJson: 4597

21:36:32.911 [main] DEBUG com.cnebrera.uc3.tech.lesson9.Measurement - [PerformanceSerialization] meanProto: 224

21:36:32.911 [main] DEBUG com.cnebrera.uc3.tech.lesson9.Measurement - [PerformanceSerialization] meanKryo: 1365

22:21:24.937 [main] DEBUG com.cnebrera.uc3.tech.lesson9.Measurement - [PerformanceDeSerialization] meanJaxb: 17741

22:21:24.938 [main] DEBUG com.cnebrera.uc3.tech.lesson9.Measurement - [PerformanceDeSerialization] meanJson: 6417

22:21:24.938 [main] DEBUG com.cnebrera.uc3.tech.lesson9.Measurement - [PerformanceDeSerialization] meanProto: 1219

22:21:24.938 [main] DEBUG com.cnebrera.uc3.tech.lesson9.Measurement - [PerformanceDeSerialization] meanKryo: 1541

23:31:20.723 [main] DEBUG com.cnebrera.uc3.tech.lesson9.Measurement - [PerformanceSerializationAndDeserialization] meanJaxb: 26682

23:31:20.724 [main] DEBUG com.cnebrera.uc3.tech.lesson9.Measurement - [PerformanceSerializationAndDeserialization] meanJson: 10750

23:31:20.725 [main] DEBUG com.cnebrera.uc3.tech.lesson9.Measurement - [PerformanceSerializationAndDeserialization] meanProto: 1444

23:31:20.727 [main] DEBUG com.cnebrera.uc3.tech.lesson9.Measurement - [PerformanceSerializationAndDeserialization] meanKryo: 3080

23:54:42.143 [main] DEBUG com.cnebrera.uc3.tech.lesson9.Measurement - [PerformanceSizeSerialization] Jaxb: 388 bytes

23:54:42.144 [main] DEBUG com.cnebrera.uc3.tech.lesson9.Measurement - [PerformanceSizeSerialization] Json: 685 bytes

23:54:42.145 [main] DEBUG com.cnebrera.uc3.tech.lesson9.Measurement - [PerformanceSizeSerialization] Proto: 56 bytes

23:54:42.146 [main] DEBUG com.cnebrera.uc3.tech.lesson9.Measurement - [PerformanceSizeSerialization] Kryo: 1024 bytes