


G1962 - Programación

Grado en Ingeniería Civil
Problemas 6

Javier González Villa
(19 de diciembre de 2025)

Licencia: Creative Commons BY-NC-SA 4.0 Internacional 

Librerías

Ejercicio 1:

Implementar un código que genere una lista de 1000 números aleatorios que sigan una distribución normal $N(5, 2)$ mediante la librería estándar RANDOM o mediante la librería NUMPY. Posteriormente, comprobar usando la librería STATISTICS que los datos generados son correctos, es decir, que la media y la desviación típica corresponden con las solicitadas en la distribución normal previa. Por último, representar los datos de manera visual mediante una gráfica apropiada (histograma, boxplot, lineal, etc.) de tu elección utilizando la librería MATPLOTLIB.

Ejercicio 2:

Implementar un código que retorne la fecha y hora actuales, que posteriormente espere 10 segundos y que vuelva a pintar la fecha y hora actuales. Una vez realizado esa prueba inicial se quiere implementar una función denominada **reloj()** que cuando sea ejecutada muestre de manera continua por pantalla la hora actual, con actualizaciones cada segundo.

Ejercicio 3:

Implementar dos códigos iguales que generen una lista con 100 millones de valores numéricos iguales y los sumen utilizando un bucle sencillo. El primero de ellos deberá ser implementado mediante la creación de listas tradicional y el otro mediante el uso de la librería estándar ITER-TOOLS. Posteriormente medir los tiempos de ejecución de cada uno de los códigos y comprobar si es cierto que la librería es capaz de realizar dichos cálculos en menor tiempo.

Ejercicio 4:

Implementar un código, que mediante el uso de las librerías estándar de Python y la librería MATPLOTLIB, represente en una gráfica de tu elección unos datos sintéticos generados de manera aleatoria o sistemática. Es recomendable consultar tanto la teoría como la página de documentación de MATPLOTLIB a la hora de elegir cual trabajar y entender como funciona.

Ejercicio 5:

Mediante la utilización de la librería PANDAS almacenar en un **Dataframe** una tabla con 10 estudiantes ficticios recogiendo los siguientes campos: 1) Nombre, 2) Apellidos, 3) Sexo, 4) DNI, 5) Nota Selectividad y 6) Nota Programación. Posteriormente introduce una columna adicional

a la tabla que indique si el alumno aprueba o suspende la asignatura de Programación en dependencia a su nota media teniendo como criterio mínimo de aprobado un 5. Tras la creación de la tabla ordenar los elementos de manera ascendente por el campo *Nota Selectividad*. Posteriormente calcular la nota media, máxima y mínima de programación para cada grupo divididos por sexo.