

Turno 01:

Desarrolle un programa completo en Python que permita generar una sucesión de **10000 números enteros aleatorios, usando como semilla del generador al valor 29 (es decir, `random.seed(29)`)**. Los valores de cada uno de esos 10000 números deben estar **entre 1 y 25000 (incluidos ambos - DEBE usar `random.randint(1, 25000)` para generar cada uno de estos números)**. A partir de esa sucesión, el programa debe:

1. Determinar cuántos de esos números eran mayores o iguales a 1 pero menores que 10000, cuántos eran mayores o iguales que 10000 y menores que 20000, y cuántos eran mayores o iguales que 20000.
2. Determinar cuántos de los números generados eran divisibles por 6.
3. Determinar el **menor** entre todos los números generados.
4. Determinar el promedio **entero** entre todos los números generados. Aclaración: NO se pide el promedio redondeado, sino el promedio truncado, sin decimales.

Turno 02:

Desarrolle un programa completo en Python que permita generar una sucesión de **15000 números enteros aleatorios, usando como semilla del generador al valor 31 (es decir, `random.seed(31)`)**. Los valores de cada uno de esos 15000 números deben estar **entre 1 y 55000 (incluidos ambos - DEBE usar `random.randint(1, 55000)` para generar cada uno de estos números)**. A partir de esa sucesión, el programa debe:

1. Determinar cuántos eran mayores que 5000 y divisibles por 3, cuántos eran mayores que 5000 y divisibles por 5 y cuántos eran mayores que 5000 y divisibles por 9.
2. Determinar el promedio **entero** entre los números mayores a 5000 y divisibles por 3. Aclaración: NO se pide el promedio redondeado, sino el promedio truncado, sin decimales.
3. Determinar el **mayor** entre todos los números generados **que sean impares**.
4. Determinar el porcentaje **entero** que la cantidad de **números mayores a 5000 y divisibles por 5** representa sobre la cantidad total de números. *Observación: en el cálculo de este porcentaje, haga primero la multiplicación que corresponda, y luego la división.*

Turno 03:

Desarrolle un programa completo en Python que permita generar una sucesión de **13000 números enteros aleatorios, usando como semilla del generador al valor 47 (es decir, `random.seed(47)`)**. Los valores de cada uno de esos 13000 números deben estar **entre 1 y 33000 (incluidos ambos - DEBE usar `random.randint(1, 33000)` para generar cada uno de estos números)**. A partir de esa sucesión, el programa debe:

1. Determinar cuántos eran mayores o iguales a 1 pero menores a 15000 y además eran divisibles por 4; cuántos eran mayores o iguales a 15000 pero menores que 22000 pero además eran pares, y cuántos eran mayores o iguales a 22000 pero además eran divisibles por 7.
2. Determinar la suma de todos los números generados que estén entre 4000 y 11000 (incluidos ambos).
3. Determinar el **menor** entre todos los números generados **cuyos últimos dos dígitos sean iguales a 23** (es decir, aquellos cuyo resto al dividir por 100 es igual a 23).
4. Determinar el porcentaje **entero** que la cantidad de **números entre 4000 y 11000 (incluidos ambos)** representa sobre la cantidad total de números. *Observación: en el cálculo de este porcentaje, haga primero la multiplicación que corresponda, y luego la división.*

Turno 04:

Desarrolle un programa completo en Python que permita generar una sucesión de **37000 números enteros aleatorios, usando como semilla del generador al valor 23 (es decir, `random.seed(23)`)**. Los valores de cada uno de esos 37000 números deben estar **entre -30000 y 30000 (incluidos ambos - DEBE usar `random.randint(-30000, 30000)` para generar cada uno de estos números)**. A partir de esa sucesión, el programa debe:

1. Determinar cuántos eran negativos y además eran pares; cuántas veces entró el cero, y cuántos eran positivos pero además eran impares.
2. Determinar el promedio **entero** de todos los números generados que sean positivos pero menores o iguales a 5000 (no incluya al cero en la cuenta). Aclaración: NO se pide el promedio redondeado, sino el promedio truncado, sin decimales.
3. Determinar el **mayor** entre todos los números generados que sean **negativos pares**.
4. Determinar el porcentaje **entero** que la cantidad de **números entre -10000 y 10000 (incluidos ambos)** representa sobre la cantidad total de números. *Observación: en el cálculo de este porcentaje, haga primero la multiplicación que corresponda, y luego la división.*

Turno 05:

Desarrolle un programa completo en Python que permita generar una sucesión de **40000 números enteros aleatorios, usando como semilla del generador al valor 53 (es decir, `random.seed(53)`)**. Los valores de cada uno de esos 40000 números deben estar **entre -10000 y 50000 (incluidos ambos - DEBE usar `random.randint(-10000, 50000)` para generar cada uno de estos números)**. A partir de esa sucesión, el programa debe:

1. Determinar **la suma** de todos los números que eran negativos; también determinar **cuántos números** eran mayores o iguales a cero pero menores que 20000, y **cuántos números** eran mayores o iguales que 20000 pero además tenían el último dígito igual a 8 o igual a 9 (el resto de dividir por 10 debe ser 8 o 9).
2. Determinar el promedio **entero** de todos los números generados que sean mayores a cero pero que además sean divisibles por 6. Aclaración: NO se pide el promedio redondeado, sino el promedio truncado, sin decimales.
3. Determinar el **menor** entre todos los números generados que sean **negativos impares**.
4. Determinar el porcentaje **entero** que la cantidad de **números negativos impares** representa sobre la cantidad total de números. *Observación: en el cálculo de este porcentaje, haga primero la multiplicación que corresponda, y luego la división.*

Turno 06:

Desarrolle un programa completo en Python que permita generar una sucesión de **35000 números enteros aleatorios, usando como semilla del generador al valor 31 (es decir, `random.seed(31)`)**. Los valores de cada uno de esos 35000 números deben estar **entre -15000 y 25000 (incluidos ambos - DEBE usar `random.randint(-15000, 25000)` para generar cada uno de estos números)**. A partir de esa sucesión, el programa debe:

1. Determinar **la suma** de todos los números que eran negativos y divisibles por 8; también determinar **cuántos números** eran mayores o iguales a cero pero menores que 10000 y que además eran impares, y **cuántos números** eran mayores o iguales que 10000 pero además eran divisibles por 7.
2. Determinar el promedio **entero** de todos los números generados que eran mayores que -5000 pero que además sean divisibles por 4. Aclaración: NO se pide el promedio redondeado, sino el promedio truncado, sin decimales.
3. Determinar el **mayor** entre todos los números generados que estén comprendidos entre -2000 y 5000 (incluidos ambos) y que sean también divisibles por 3.

4. Determinar el porcentaje *entero* que *la cantidad de números negativos y divisibles por 8* representa sobre la cantidad total de números. *Observación: en el cálculo de este porcentaje, haga primero la multiplicación que corresponda, y luego la división.*

Turno 07:

Desarrolle un programa completo en Python que permita generar una sucesión de *27000 números enteros aleatorios, usando como semilla del generador al valor 37 (es decir, random.seed(37))*. Los valores de cada uno de esos 27000 números deben estar *entre -20000 y 30000 (incluidos ambos - DEBE usar random.randint(-20000, 30000) para generar cada uno de estos números)*. A partir de esa sucesión, el programa debe:

1. Determinar *cuántos de esos números* son mayores o iguales que -20000 pero menores que -5000; también determinar *cuántos números* son mayores o iguales a -5000 pero menores que 15000, y *cuántos números* son mayores o iguales que 15000 pero además son divisibles por 9.
2. Determinar el promedio *entero* de todos los números generados que sean mayores o iguales a 1000 pero que además tengan su último dígito igual a 4 o a 6 (es decir, el resto de dividir por 10 debe ser 4 o 6). Aclaración: NO se pide el promedio redondeado, sino el promedio truncado, sin decimales.
3. Determinar el *mayor* entre todos los números generados que sean *positivos impares*, pero que además tengan su último dígito diferente de 1 (es decir, además de ser impar, el resto de dividir por 10 debe ser diferente de 1).
4. Determinar el porcentaje *entero* que la cantidad de *números divisibles por 7* representa sobre la cantidad total de números. *Observación: en el cálculo de este porcentaje, haga primero la multiplicación que corresponda, y luego la división.*