

5. Sesión nº 5

Objetivos: Usar funciones en python. Refuerzo de bucles. Más sobre entrada/salida.

1. Observa el siguiente programa que utiliza dos funciones: una para encapsular la entrada de datos y otra para calcular y mostrar la nota final.

```
def pide_notas():
    prueba1 = float(input("Nota 1ª prueba: "))
    prueba2 = float(input("Nota 2ª prueba: "))
    practicas = float(input("Nota prácticas: "))
    return prueba1, prueba2, practicas

def muestra_nota(test1, test2, practicas):
    final = test1*0.35 + test2*0.35 + practicas*0.3
    print("Nota final: {:.2f}".format(final))

nota1, nota2, nota3 = pide_notas()
muestra_nota(nota1, nota2, nota3)
```

2. El siguiente programa repite la entrada y visualización de notas hasta que se indica que se quiere finalizar

```
def pide_notas():
    prueba1 = float(input("Nota 1ª prueba: "))
    prueba2 = float(input("Nota 2ª prueba: "))
    practicas = float(input("Nota prácticas: "))
    return prueba1, prueba2, practicas

def muestra_nota(test1, test2, practicas):
    final = test1*0.35 + test2*0.35 + practicas*0.3
    print("Nota final: {:.2f}".format(final))

repetir = "si"
while (repetir != "no"):
    nota1, nota2, nota3 = pide_notas()
    muestra_nota(nota1, nota2, nota3)
    repetir = input("Quiere seguir (si/no) ")
    repetir = repetir.lower()
else:
    print("\nFin del programa")
```

3. Modifica el programa anterior de tal manera que alguna entrada de un valor negativo como nota termina el bucle (claro, sin que esta última nota participe en la media).
4. Escribe un programa para calcular la secuencia de Collatz de un número entero positivo a_0 , donde la secuencia es: $a_{n+1} = 1/2 \cdot a_n$, si a_n es par, y $a_{n+1} = 3 \cdot a_n + 1$, si a_n es impar. La secuencia termina si llegamos al número 1. Por ejemplo, para $a_0 = 3$ tenemos la secuencia: 3, 10, 5, 16, 8, 4, 2, 1.

Nota: la conjetura de Collatz supone que, para cualquier valor de n la secuencia siempre llega a 1. La conjetura de Collatz aún no ha podido ser demostrada¹.

5. Sabemos en, en España, la altura de la población joven masculina sigue una distribución normal (media 172 cm, desviación estándar 8.1 cm). Queremos calcular el porcentaje de dicha población que supera 185 cm de estatura. Para ello usaremos el método de Montecarlo: generamos puntos de datos que sigan dicha distribución, y contamos cuantos de dichos puntos superan la altura dada (185 cm). Escribe una función para calcular dicho porcentaje.

Nota1: consideraremos que 10000 puntos es suficiente para una estimación aproximada.

Nota2: la librería `random` proporciona una función llamada `gauss` que puede ser utilizada para generar los datos. Ver la documentación de python²

6. Modificar el programa del apartado anterior de modo que se pida que el usuario introduzca por teclado los datos para el cálculo: media, desviación estándar y el umbral de altura para calcular el porcentaje de población que se encuentra por encima del mismo.

¹https://es.wikipedia.org/wiki/Conjetura_de_Collatz

²<https://docs.python.org/3/library/random.html>