



## **EJERCICIOS 4 FUNCIONES Y PROCEDIMIENTOS**

- 1) Crea un procedimiento EscribirEspaciado, que reciba como parámetro un texto y lo escriba con un espacio adicional tras cada letra. Por ejemplo, “Hola, tú” se escribiría “H o l a , t ú “.
- 2) Crea una función Iniciales, que devuelva una cadena formada por las iniciales de la frase que se indique como parámetro (primera letra y la letra que haya tras cada espacio; por ejemplo, para “Nacho Cabanes” devolvería “NC”).
- 3) Crea una función Contiene, que reciba una cadena y una (posible) subcadena, y devuelva “verdadero” o “falso”, según si la primera contiene a la segunda (como “Nacho” contiene “ac”) o no la contiene, (como “Aurora” no contiene “sol”).
- 4) Crea una función que pida su nombre al usuario y lo escriba al revés y lo convierta a MAYUSCULA si estaba en minúscula y Viceversa (de la última letra a la primera: a partir de “Nacho” escribiría “OHCAn”).
- 5) Crea una función que encripte una contraseña utilizando la hora fecha y vocal para reemplazar por cada letra de la contraseña.
- 6) Crea una función que desemcripte una contraseña utilizando el formato de encriptación de la pregunta anterior.
- 7) Crea un programa que pida su nombre al usuario y diga qué vocales contiene (en orden y sin repetir: por ejemplo, para “Aurora” deberá responder “aou”).

- 8) Cree una función que retorne la palabra de una frase que tenga mayor longitud.
- 9) Escribir una función `sum()` y una función `multip()` que sumen y multipliquen respectivamente todos los números de una lista. Por ejemplo: `sum([1,2,3,4])` debería devolver 10 y `multip([1,2,3,4])` debería devolver 24.
- 10) Definir un procedimiento `procedimiento()` que tome una lista de números enteros e imprima un histograma en la pantalla. Ejemplo: `procedimiento([4, 9, 7])` debería imprimir lo siguiente: \*\*\*\* \*\*\*\*\*  
\*\*\*\*\*