

EXAMEN DE PROGRAMACIÓN PHP

1ª EVALUACIÓN – 2º DWES

- 1- Se denomina “distancia Hamming” de dos cadenas de igual longitud al número de caracteres diferentes que tienen. Por ejemplo, la distancia Hamming de “Carmen” y “Carmel” es 1.

Por otra parte, es importante saber si esta distancia debe calcularse mediante una comparación case-sensitive o case-insensitive. En el primer caso, la distancia entre “Carmen” y “CARMEN” sería 5, y en el segundo, sería 0.

Haz una función denominada **distanciaHamming** que reciba tres parámetros. Los dos primeros se llamarán **stringA** y **stringB** y serán las cadenas a comparar. El tercero, será un booleano, **caseSensitive**, que será un parámetro opcional y **valdrá por defecto true**. La función deberá comparar la longitud de ambas cadenas, y si no fuera igual esa longitud, devolverá el valor -1.

1.5 puntos

Esta función, estará ubicada en una librería denominada **librería_examen.php** y será invocada por un programa, **ejercicio1.php**.

El programa deberá:

- pedir las dos cadenas por un formulario de entrada (este código está adjunto al final) y llamarse a sí mismo utilizando el método POST
- evitar un posible ataque XSS vía formulario
- invocar a la función dos veces con ambas cadenas, primero con el parámetro **caseSensitive** a false, y luego sin él, y mostrar el valor devuelto, salvo que el resultado fuera -1, en cuyo caso emitirá un mensaje de error.

1.5 puntos

CÓDIGO DEL FORMULARIO DE ENTRADA DEL EJERCICIO 1:

```
<form>
  <h3>Introduce dos cadenas y te calculo la distancia Hamming entre
  ellas</h3>
  <div>
    <label for="cadena1">Cadena 1:</label>
    <input type="text" id="cadena1" name="cadena1" required>
  </div>
  <div>
    <label for="cadena2">Cadena 2:</label>
    <input type="text" id="cadena2" name="cadena2" required>
  </div>
  <div>
    <button type="submit">Enviar</button>
  </div>
</form>
```

- 2- Un conjunto es una estructura de datos que contiene una colección de valores sin repeticiones ni orden entre ellos. No existen como tal en PHP, pero vamos a simularlos con la creación de una clase.

Crea en el archivo **Conjunto.php** la clase **Conjunto** que contenga como atributos (privados):

- a. Un array de enteros llamada **set**
- b. Un tamaño máximo del conjunto, llamado **maxItems**
- c. Un número de elementos actuales en el conjunto, llamado **items**

Y como métodos:

- a. El **constructor**, que iniciará el array a vacío, los ítems a 0 y el tamaño será su único parámetro
- b. El **destructor**
- c. El método para **pasar a string** el contenido del conjunto (un conjunto debe lucir así de cara al exterior: {1, 5, 2, 8} por ejemplo)
- d. Un método **incluir**, que añada un elemento al conjunto (en caso de que no estuviera ya)
- e. Un método **incluido**, que tenga como parámetro un elemento y devuelva true o false si ese está o no incluido en el conjunto, respectivamente
- f. Un método **intersección** que utilice otro conjunto como parámetro, y devuelva los elementos en común (otro conjunto)
- g. Un método **union** que utilice otro conjunto como parámetro, y devuelva el conjunto que contiene a los elementos de uno y otro sin repetición
- h. Un método **diferencia** que utilice otro conjunto como parámetro, y devuelva el conjunto que contiene a los elementos de no tienen en común uno y otro

4 puntos