

EXAMEN DE PROGRAMACIÓN PHP

1^a EVALUACIÓN – 2º DWES – Modelo B

- 1- Crea una página con formulario denominada **formulario.php** (el código está en la parte inferior) que solicite dos cadenas de entrada: una frase y una serie de caracteres, que son requeridos. Este formulario deberá enviar de manera segura los datos de entrada a otro denominado **proceso.php**

En **proceso.php**, crea una función de nombre **estanIncluidos** que recibirá dos parámetros requeridos de tipo string denominados **frase** y **caracteres**, y otro parámetro booleano **caseSensitive** (valor por defecto, true. La función devolverá a su vez otro booleano:

- a. true si todos los **caracteres** están incluidos en **frase**, da igual si sueltos, unidos y en qué orden
- b. false si no es así
- c. si el parámetro **caseSensitive** fuera false, dará igual si hay diferencias mayúsculas/minúsculas, pero es un parámetro opcional, que por defecto valdrá true
- d. Debes indicar el tipo de parámetros y valor devuelto por la función

proceso.php deberá descargar los datos enviados por el formulario, evitar un posible ataque XSS, invocar a la función con el parámetro caseSensitive a false en un caso y sin él en otro y mostrar los resultados (con echo). Por ejemplo:

estanIncluidos("Murcielago","AEIOU") → devolverá false

estanIncluidos("Murcielago","AEIOU", false) → devolverá true

3.5 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">Dame una frase:</label>
    <input type="text" id="cadena1" name="cadena1" required>
  </div>
  <div>
    <label for="cadena2">Dame una frase:</label>
    <input type="text" id="cadena2" name="cadena2" required>
  </div>
  <div>
    <button type="submit">Enviar</button>
  </div>
</form>
```

2- Crea una clase **Pila** que contenga como atributos (privados):

- a. Un array llamada **pila**
- b. Un tamaño máximo de la pila, llamado **longitud**
- c. Un número de elementos en la pila, llamado **elementos**

Y como métodos:

- a. El **constructor**, que iniciará el array pila a vacío, los ítems a 0 y el tamaño será requerido
- b. Un método **push**, que añada un elemento a la pila por un extremo. Si la pila estuviera ya llena, devolverá null. No devolverá nada en otro caso
- c. Un método **pop** que extraiga un elemento de la pila. Devolverá ese elemento, salvo que la pila esté vacía, en cuyo caso, devolverá null
- d. Un método **getElementos** para obtener el número de elementos almacenado en la pila en un momento dado
- e. Un método **para pasar a String** que obtenga la pila completa en un momento dado

Comprueba mediante una serie de instrucciones que generen salida (echo), que todo funciona correctamente:

1. Crea una pila de tamaño 10
2. Inserta un 5 en la pila (y controla que pueda estar llena)
3. Inserta un 3 en la pila (y controla que pueda estar llena)
4. Muestra el tamaño de la pila
5. Extrae un elemento de la pila (y controla que pueda estar vacía)
6. Muestra el tamaño de la pila
7. Muestra la pila completa

3.5 puntos

MODO DE ENTREGA:

El examen se entregará comprimido en un archivo con los apellidos+nombre de cada alumnx y se subirá a la Moodle, que quedará habilitada para este fin a las 20:30 horas, debiéndose permanecer en el aula hasta este momento