

Ac3.4.- Utilización de Arrays

- a) Codificar un programa que realice la fusión de dos arrays de 20 números ordenados. Implementa las funciones: cargar (genera un array con números aleatorios – rand()), ordenar (ordena los elementos de un array) y mezclar (concatena dos arrays).
- b) Realizar un programa en PHP que nos permita calcular la letra del NIF en base al DNI de una persona. Para calcular la letra dividiremos el DNI por 23 y dependiendo del resto asignaremos la letra correspondiente (buscar la asociación correspondiente).
- c) Realiza un programa PHP que utilice foreach para mostrar todos los valores del array \$_SERVER en una tabla con dos columnas. La primera columna debe contener el nombre de la variable, y la segunda su valor.
- d) Modifica el último programa realizado para obtener la fecha en castellano, de forma que prescindamos de las estructuras switch-case y en su lugar almacenemos los nombres de los meses y días de la semana en sendos arrays. El programa deberá calcular la fecha en castellano en una función, y mostrarla por pantalla desde el programa principal.
- e) Crea un array asociativo de países y capitales inicializándolo con los principales países y capitales europeos. Llama al array “mapa”.

Visualiza el array con la función print_r.

Muestra la capital de Italia.

Averigua si está contenida Rusia.

Recorre el array mediante un bucle foreach para visualizar todo su contenido.

Elimina a Francia de la lista.

Vuelve a mostrar el contenido del array con foreach.

Muestra el país cuya capital es Lisboa.

- f) Escribe una función que reciba un array de números, y un número, el límite. La función tiene que devolver un nuevo array que contenga solo los elementos del array original menores que el límite.

g) Escribe una función que reciba una cadena y compruebe si es un palíndromo. Realiza varias llamadas a la función para comprobar su comportamiento.

h) Realiza un programa que sirva para mostrar el presente, pasado y futuro de un **verbo regular**. Consideramos los verbos de la primera conjugación ("-ar").

Carga un array con diez verbos regulares, terminados en "-ar".

Codifica en tres arrays las terminaciones del presente, pasado y futuro de indicativo.

El programa seleccionará uno de los verbos aleatoriamente y mostrará los tres tiempos verbales.

Los resultados se obtienen uniendo la raíz del verbo con cada valor del array que le corresponde a su tiempo verbal.

\$presente=array("o", "as", "a", "amos", "áis", "an")

Por ejemplo:

Presente del verbo amar: **am-o**, **am-as**, **am-a**, **am-amos**, **am-áis**, **am-an**.

Pasado: **am-é**, **am-aste**, **am-ó**, **am-amos**, **am-asteis**, **am-aron**.

Futuro: **am-aré**, **am-arás**, **am-ará**, **am-aremos**, **am-aréis**, **am-arán**.

Utiliza funciones y estructuras lo más eficientemente posible.

Utiliza un editor o un entorno de desarrollo para crear los programas requeridos y ubícalos en la carpeta correspondiente del servidor web, de forma que puedas acceder a ellos con un navegador.