# A1.2 - Arrays i funcions

## Exercici 1. Factorial d’un array

Crea una funció ***factorialArray()*** que li passes un array de nombres i retorna un altre array amb el [factorial](https://ca.wikipedia.org/wiki/Factorial) de cada nombre.

La funció ha de retornar *false* en cas que el paràmetre rebut no sigui un array o bé algun dels valors de l’array no sigui un nombre.

**AMPLIACIÓ**: Utilitza recursivitat per calcular el factorial d’un nombre.

El codi que hauríem d’utilitzar per aquest exercici seria el següent:

<?php

function factorialNumero($numero) {

/\*\*

\* Iniciem la variable factorial

\* i el comptador amb 1 ja que sinó

\* el resultat seria sempre 0

\*/

$factorial = 1;

for ($i = 1; $i <= $numero; $i++) {

$factorial = $factorial \* $i;

}

return $factorial;

}

function factorialArray($arrayNumeros) {

/\*\*

\* Hem de saber primer si el paràmetre

\* és un array per fer el procés

\*/

if (is\_array($arrayNumeros)){

$arrayFactorials = array();

for ($i = 0; $i < count($arrayNumeros); $i++) {

/\*\*

\* Hem de saber primer si el valor de

\* la casella és un número per fer el

\* procés

\*/

if (is\_numeric($arrayNumeros[$i])) {

/\*\*

\* Apliquem la funció creada anteriorment

\* per cada valor de cada casella

\*/

$arrayFactorials[$i] = factorialNumero($arrayNumeros[$i]);

} else {

return false;

}

}

return $arrayFactorials;

} else {

return false;

}

}

/\*\*

\* Amb aquest mètode, podem passar

\* un array a String separant els

\* valors per espais

\*/

function arrayToString($array) {

$cadenaArray = "[ ";

foreach( $array as $valor ) {

$cadenaArray .= $valor . " ";

}

$cadenaArray .= "]";

return $cadenaArray;

}

echo "<h1><u>Factorial d'un array</u></h1>";

$arrayInicial = array(0, 8, 10, 12);

$arrayFactorialNumeros = factorialArray($arrayInicial);

echo "<h2>Array de números</h2>";

echo var\_dump($arrayInicial);

echo "<h2>Array de números (convertit a String)</h2>";

echo arrayToString($arrayInicial);

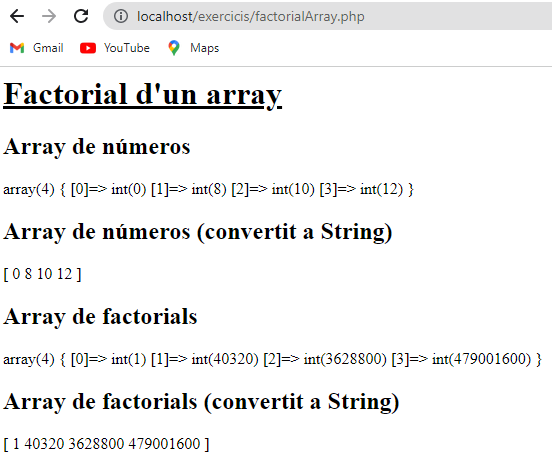
echo "<h2>Array de factorials</h2>";

echo var\_dump($arrayFactorialNumeros);

echo "<h2>Array de factorials (convertit a String)</h2>";

echo arrayToString($arrayFactorialNumeros);

?>



Aquí veiem com el codi ens funciona.

## Exercici 2. Arrays multidimensionals

⚠️ Les matrius no han de ser quadrades

1. Crea un funció ***creaMatriu(n)*** que generi una matriu quadrada (nxn, n files i n columnes) a partir del valor n passat per paràmetre.

Aquesta matriu ha de tenir:

* Un \* a cada casella de la diagonal.
* Un nombre aleatori entre 10 i 20 a les caselles de sota la diagonal.
* El valor de suma de la fila i la columna a les caselles de sobre la diagonal.

| \* | 1 | 2 | 3 |
| --- | --- | --- | --- |
| 11 | \* | 3 | 4 |
| 15 | 16 | \* | 5 |
| 19 | 12 | 13 | \* |

1. Crea una funció ***mostraMatriu(...)*** que retorni la matriu en forma de taula HTML. La funció retornarà un String.
2. Finalment, crea un funció ***transposaMatriu(...)*** que rep una matriu i retorni una altra matriu amb el valor de les files per les columnes intercanviats

| \* | 1 | 2 | 3 |
| --- | --- | --- | --- |
| 11 | \* | 3 | 4 |
| 15 | 16 | \* | 5 |
| 19 | 12 | 13 | \* |

| \* | 11 | 15 | 19 |
| --- | --- | --- | --- |
| 1 | \* | 16 | 12 |
| 2 | 3 | \* | 13 |
| 3 | 4 | 5 | \* |

Per fer aquests exercicis serà necessari el següent codi:

<?php

function creaMatriu (int $n) {

/\*\*

\* Iniciem les variables que utilitzarem,

\* entre elles la matriu que retornarem

\*/

$matriu = array();

$fila = 0;

$columna = 0;

/\*\*

\* Amb aquests dos bucles analitzarem les files i,

\* dins d'aquestes, les columnes o caselles per

\* cada fila

\*/

for ($fila = 0; $fila < $n; $fila++) {

for ($columna = 0; $columna < $n; $columna++) {

/\*\*

\* Si el número de files equival al de columnes,

\* dibuixarem la diagonal que separarà

\* el contingut de l'array

\*/

if ($fila == $columna) {

$matriu[$fila][$columna] = "\*";

}

/\*\*

\* En cas de que el número de files sigui

\* inferior al de les columnes, equivaldrà

\* a la meitat superior de l'array i per tant,

\* col·locarem la suma d'aquestes dos variables

\*/

else if ($fila < $columna) {

$matriu[$fila][$columna] = $fila + $columna;

}

/\*\*

\* En cas de que el número de files sigui

\* major al de les columnes, equivaldrà a la

\* meitat inferior de l'array i per tant,

\* col·locarem els números aleatoris

\*/

else if ($fila > $columna) {

$matriu[$fila][$columna] = rand(10,20);

}

}

}

return $matriu;

}

function mostraMatriu($matriu) {

/\*\*

\* Iniciem la variable on guardarem

\* la taula completa amb l'etiqueta

\* principal table i amb l'atribut

\* border perquè es vegi bé

\*/

$taula = '<table border="1">';

/\*\*

\* En aquests dos bucles, afegim

\* l'etiqueta <tr> per cada fila

\* analitzada i l'etiqueta <th>

\* per cada columna dins d'aquesta.

\*/

foreach($matriu as $fila) {

$taula .= "<tr>";

foreach($fila as $columna) {

$taula .= "<th>" . $columna . "</th>";

}

$taula .= "</tr>";

}

/\*\*

\* Tanquem l'etiqueta de la taula

\*/

$taula .= "</table>";

return $taula;

}

function transposaMatriu($matriu) {

$files = 0;

$columnes = 0;

$matriuGirada = array();

/\*\*

\* Per adaptar-lo tant si la matriu entrada per

\* paràmetre és quadrada o no, comptarem tant

\* les files com les columnes d'aquesta

\*/

$files = count($matriu);

for ($fila = 0; $fila < $files; $fila++) {

$columnes = count($matriu[$fila]);

for($columna = 0; $columna < $columnes; $columna++) {

if ($fila == $columna) {

$matriuGirada[$fila][$columna] = "\*";

}

/\*\*

\* En aquests dos condicionals s'agafa el valor de la

\* matriu entrada per paràmetre i es guarda en la

\* posició exactament invertida de la nova que hem creat.

\* Ho fem girant les files i les columnes d'aquesta.

\*/

else if ($fila > $columna) {

$matriuGirada[$columna][$fila] = $matriu[$fila][$columna];

}

else if ($fila < $columna) {

$matriuGirada[$columna][$fila] = $matriu[$fila][$columna];

}

}

}

return $matriuGirada;

}

echo "<h1><u>Crear, mostrar i transposar matriu</u></h1>";

$matriuCreada = creaMatriu(4);

$matriuInvertida = transposaMatriu($matriuCreada);

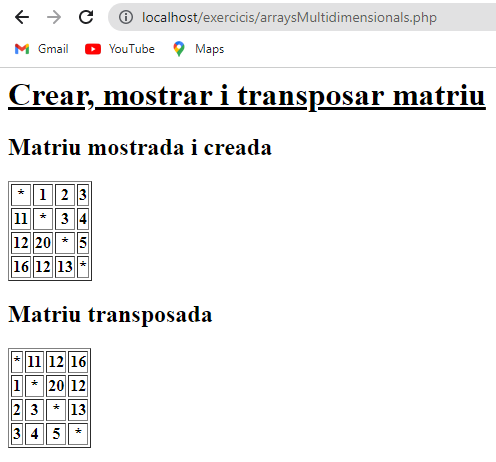
echo "<h2>Matriu mostrada i creada</h2>";

echo mostraMatriu($matriuCreada);

echo "<h2>Matriu transposada</h2>";

echo mostraMatriu($matriuInvertida);

?>



Aquí veiem que el codi ens funciona.

ENLLAÇ A LA SUBCARPETA DEL GITHUB (PRÀCTICA):

<https://github.com/janEstrada24/2DAW/tree/main/DWES/UF1/A2>

ENLLAÇ A LA SUBCARPETA (PROGRAMES PHP):

<https://github.com/janEstrada24/2DAW/tree/main/DWES/UF1/A2/PHP>

Els programes s’anomenen **factorialArray.php** i **arraysMultidimensionals.php**