

Para todos los ejercicios, realice el Diagrama de Estructura correspondiente.

1. Realice una función que indique si un elemento dado se encuentra en un arreglo de caracteres.
2. Realice un procedimiento / función que inserte un caracter en un arreglo ordenado alfabéticamente, conservando el orden.
3. Codifique funciones que devuelvan el máximo, la suma total y el promedio de las longitudes de los strings en un arreglo dado. ¿Se modificarían las soluciones si el arreglo estuviera ordenado por longitud?

Ejemplo: `array('a', 'bcde', 'fg', 'hijk', 'lmn');` → máximo: 4; total: 14; promedio: 2,8.

4. Codifique funciones que devuelvan el máximo, la suma total y el promedio de las longitudes de los strings en una matriz de dos dimensiones dada, utilizando las funciones realizadas en el ejercicio anterior.
5. Ordene un string, utilizándolo como arreglo de caracteres, por los métodos de
 1. selección
 2. inserción
 3. burbujeo

No utilice arreglos ni strings auxiliares.

6. Realice un procedimiento / función que determine si un arreglo es capicúa.
7. Realice un procedimiento que muestre en pantalla el contenido de una matriz de dos dimensiones, de manera que los valores de las columnas queden alineados y que todas las columnas tengan el ancho dado por el elemento más largo en la matriz. Por ejemplo, dado:

```
array(
    array(3, 'aaa', 2),
    array('x', 'y', 0),
    array('abcde', 45, 0),
    array(1, 7, 60)
);
```

se debe mostrar:

```
3    aaa    2
x    y      0
abcde 45    0
1    7      60
```

Ayudas:

- La función `str_pad($var, $long)` rellena `$var` con espacios en blanco, para que quede de longitud `$long`. Ejemplo: `str_pad("cadena", 9) → "cadena "`.
- Puede reutilizar una de las funciones implementadas en el Ejercicio 2.

¿Qué debería cambiar para que cada columna se muestre con el ancho dado por el elemento más largo *en la columna*?

8. Realice un procedimiento/función que construya la matriz transpuesta de una matriz de dos dimensiones dada. Compruebe que funcione correctamente, mostrando la matriz original y la transpuesta.

La transpuesta T de una matriz dada M tiene como columnas las filas de M, en el mismo orden. Ejemplo:

1	1	2	3
2	4	5	6
3	7	8	9
4	0	'A'	'B'

→

	1	2	3	4
1	1	4	7	0
2	4	5	8	'A'
3	7	8	9	'B'

9. Realice un procedimiento/función que construya una matriz de dos dimensiones a partir de un arreglo de enteros, de manera que la fila *i*, columna *j* de la matriz contenga la diferencia entre las posiciones *i* y *j* del arreglo. Compruebe que funcione correctamente, mostrando el arreglo original y la matriz resultante.

Ejemplo:

3	1	2
---	---	---

→

3 - 3	3 - 1	3 - 2
1 - 3	1 - 1	1 - 2
2 - 3	2 - 1	2 - 2

→

0	2	1
-2	0	-1
-1	1	0

10. Una secuencia de elementos en un arreglo está formada por elementos de índices consecutivos distintos de NULL. Por simplicidad, se asume que aún la última secuencia en el arreglo está terminada por un NULL, es decir que el último elemento del arreglo va a ser siempre NULL.

La longitud de una secuencia está dada por la cantidad de elementos que contiene.

Por ejemplo:

<pre>\$s = array(1, 'a', NULL, 2, 'b', 3, 'c', NULL, NULL, 4, NULL);</pre>	<table style="border-collapse: collapse; margin-left: 10px;"> <tr> <td style="padding: 0 5px;">0</td><td style="padding: 0 5px;">1</td><td style="padding: 0 5px;">2</td><td style="padding: 0 5px;">3</td><td style="padding: 0 5px;">4</td><td style="padding: 0 5px;">5</td><td style="padding: 0 5px;">6</td><td style="padding: 0 5px;">7</td><td style="padding: 0 5px;">8</td><td style="padding: 0 5px;">9</td><td style="padding: 0 5px;">10</td><td style="padding: 0 5px;">← índice del elemento</td> </tr> <tr> <td style="border: 1px dashed black; width: 20px; height: 20px;"></td><td style="border: 1px dashed black; width: 20px; height: 20px;"></td><td style="border: 1px dashed black; width: 20px; height: 20px;"></td><td style="border: 1px dashed black; width: 20px; height: 20px;"></td><td style="border: 1px dashed black; width: 20px; height: 20px;"></td><td style="border: 1px dashed black; width: 20px; height: 20px;"></td><td style="border: 1px dashed black; width: 20px; height: 20px;"></td><td style="border: 1px dashed black; width: 20px; height: 20px;"></td><td style="border: 1px dashed black; width: 20px; height: 20px;"></td><td style="border: 1px dashed black; width: 20px; height: 20px;"></td><td style="border: 1px dashed black; width: 20px; height: 20px;"></td><td></td> </tr> <tr> <td style="padding: 5px 0 0 20px;">0</td><td style="padding: 5px 0 0 40px;">1</td><td style="padding: 5px 0 0 60px;">2</td><td style="padding: 5px 0 0 80px;">3</td><td colspan="7"></td><td style="padding: 5px 0 0 80px;">← índice de la secuencia</td> </tr> <tr> <td style="padding: 5px 0 0 20px;">2</td><td style="padding: 5px 0 0 40px;">4</td><td style="padding: 5px 0 0 60px;">0</td><td style="padding: 5px 0 0 80px;">1</td><td colspan="7"></td><td style="padding: 5px 0 0 80px;">← longitud de la secuencia</td> </tr> </table>	0	1	2	3	4	5	6	7	8	9	10	← índice del elemento													0	1	2	3								← índice de la secuencia	2	4	0	1								← longitud de la secuencia
0	1	2	3	4	5	6	7	8	9	10	← índice del elemento																																						
0	1	2	3								← índice de la secuencia																																						
2	4	0	1								← longitud de la secuencia																																						

Codifique procedimientos/funciones que, dado un arreglo de secuencias, resuelvan los siguientes problemas:

1. construya un arreglo que contenga los índices del elemento inicial de cada secuencia. Dado \$s, el resultado sería:

| array(0, 3, 8, 9)

2. construya un arreglo que contenga los índices de las secuencias, de manera que queden en forma ordenada ascendente de acuerdo a su longitud (es decir, el primer elemento del arreglo debe tener el índice de la secuencia más corta, y el último, el de la más larga). Dado \$s, el resultado sería:

| array(2, 3, 1, 0)

3. construya un arreglo que contenga las mismas secuencias que el dado, ordenadas ascendentemente por longitud. Utilice los procedimientos/funciones realizados en los dos puntos anteriores. Dado \$s, el resultado sería:

| array(NULL, 4, NULL, 1, 'a', NULL, 2, 'b', 3, 'c', NULL)

11. Implemente un procedimiento que modifique una matriz bidimensional de valores enteros, de manera que los elementos dentro de cada fila queden ordenados ascendentemente y que las filas también queden ordenadas ascendentemente de acuerdo a la suma de sus elementos. No utilice arreglos auxiliares.

Ejemplo:

0	8	4	1	0		4	0	1	5	6
1	3	7	9	1		0	0	1	4	8
2	2	6	3	9	→	3	2	2	7	8
3	2	2	8	7		1	1	3	7	9
4	0	5	6	1		2	2	3	6	9

¿Qué cambiaría si pudiera usar arreglos auxiliares?