

## Práctico 6 Arreglos

### Arrays

1. Supongamos tienes una lista con las alturas en cm de todos los miembros de tu familia, por ejemplo [181.5, 72., 34.7, 171.3, 160.1]. Crear un array y mostrar sus atributos, el tipo de datos, tanto del array como de sus elementos. Mostrar también el total de familiares cargados en el array.
2. Crear un array de 3 dimensiones, que tenga 3 matrices de 2 filas por 4 columnas. Llenelo con ceros.
3. Crear una matriz de (4,6) con valores al azar que pertenecen al intervalo [0,1).
4. Crear un vector con un total de 25 elementos equidistantes en el intervalo [1,6].
5. Pedirle 6 números enteros al usuario y guardarlos en una lista. Crear un array de una dimensión en base a dicha lista.
6. Crear un vector con números enteros al azar entre 0 y 5. Luego reemplazar los 0 con el valor -1.
7. Dada una lista de 3 números enteros cualesquiera, y un vector con 3 números enteros cualesquiera. ¿Qué sucede si suma la lista a si misma, lista + lista, y si hace lo mismo con el vector? Haga la prueba y compare los resultados.
8. Crear una matriz de 3 x 3, con valores que van de 1 a 9.
9. Crear una matriz de 16 x 20 con números al azar, de algún tipo que le guste, distinto al tipo de dato por defecto *float64*.
10. Crear un array de 5 filas y 6 columnas, llenarlo con valores numéricos enteros, al azar entre 1 y 6. Luego, reemplazar todos los valores en la fila 5, por el valor 0.
11. Crear una función que realice la suma de dos arrays de dimensión 1 y devuelva el array resultante. Sin utilizar el operador + directamente, sino creando un algoritmo que hiciese la suma lugar a lugar. (*\*a pulmón*)
12. Crear una función que realice el producto vectorial de dos arrays de dimensión 1 y devuelva el vector resultante. Sin utilizar el operador \* directamente, sino creando un algoritmo que hiciese el producto lugar a lugar. (*\*a pulmón*)
13. Extender ahora el ejercicio anterior, a otra función que permita realizar la suma de elementos lugar a lugar, de dos arrays de dimensión 2.

14. Generar la siguiente matriz e imprimirla en pantalla:

```
1 0 0 0 0 0
0 1 0 0 0 0
0 0 1 0 0 0
0 0 0 1 0 0
0 0 0 0 1 0
0 0 0 0 0 1
```

15. Generar la siguiente matriz e imprimirla en pantalla:

```
1 1 1 1 1
0 0 0 0 0
1 1 1 1 1
0 0 0 0 0
1 1 1 1 1
```

16. Generar la siguiente matriz e imprimirla en pantalla:

```
1 0 1 0 1
0 1 0 1 0
1 0 1 0 1
0 1 0 1 0
1 0 1 0 1
```

## Variado

17. Crear un array (4, 3) de nombres inventados, determinar cuantos nombres en la segunda columna terminan con la letra *s*:
18. Pedirle al usuario las notas de 3 exámenes, para 8 alumnos diferentes. Utilizar todo lo aprendido hasta ahora, como archivos de texto, creación de funciones propias, etc. Recuerde comentar el código y ser prolijo.
- a) Guardarlos en un archivo de texto llamado notas.txt
  - b) Leer ese archivo de notas, y crear una matriz para guardar esas notas. Cada columna será una nota, cada fila representará un alumno.
  - c) Calcular la nota promedio total, la nota promedio por alumno, y la nota promedio por examen.
  - d) Determinar la mejor nota de cada alumno.
  - e) Determinar la peor nota de cada alumno.
  - f) Determinar cuál de los 3 exámenes fue donde hubo la mejor nota.
  - g) Determinar cuál de los 3 exámenes fue donde hubo la peor nota.
19. Generar una matriz de 3 notas para 500 alumnos de primer año, de la carrera de arquitectura, y realizar las mismas cuentas que en el punto anterior, pero en esta matriz ahora.