

GUÍA N°4

LENGUAJE C: FUNCIONES Y ARREGLOS

Para los siguientes enunciados, **debe** hacer **una función por cada operación**. La función `main()`, solo debe actuar de “director de orquesta”. (Se sugiere reutilizar los códigos de ejemplo usados en el apunte. Para el caso de arreglos de n valores, define una constante MAX como tamaño máximo de sus arreglos –defínale un valor 10000-, y maneje la variable n para saber hasta dónde está la información útil del arreglo).

1. Haga un programa que llene un arreglo con los números enteros del 1 al 20, y lo imprima por pantalla.

A partir del programa anterior, haga un nuevo programa que:

2. Imprima el contenido del arreglo, en forma invertida.
3. Imprima el contenido del arreglo, pero sólo de las posiciones pares (la segunda posición, la cuarta posición, etc).
4. Construya un programa que promedie los valores de un arreglo de n números, definidos por el usuario, e imprima tal valor por pantalla.
5. Construya un programa maneje un arreglo de n números, definidos por el usuario, y luego indique por pantalla si un número (también definido por el usuario), está o no dentro del arreglo.
6. Reciba un arreglo con n caracteres, y un carácter cualquiera, y elimine el carácter dentro del arreglo (debe quedar el caracter espacio " "). Por ejemplo, si el arreglo original es *A* y el carácter a eliminar es “d”:

A:

a	b	c	d	a	b	c	d	a	b	d
---	---	---	---	---	---	---	---	---	---	---

El resultado debe ser: (Las celdas vacías deben quedar con un espacio en blanco “ ”).

A:

a	b	c	a	b	c	a	b			
---	---	---	---	---	---	---	---	--	--	--

7. Suponga que posee 2 arreglos *A* y *B*, donde, *A* posee n números y *B* es de m números, Construya un arreglo *C* de $n+m$ números, donde:
 - a. Primero aparezcan los valores de *A* y luego los de *B*.
 - b. Primero aparezcan los valores de *B* y luego los de *A*.
 - c. Aparezcan intercalados los valores de *A* y *B*, vale decir: el primer valor de *A*, luego el primer valor de *B*, después, el segundo valor de *A*, y luego el segundo valor de *B*, (recuerde que n y m no tienen por qué ser iguales).

- d. Ahora suponga que **A** y **B** son arreglos ordenados ascendentemente. Genere el nuevo arreglo **C** que esté ordenado de la misma forma.
 - e. Suponga que **A** está ordenado en forma ascendente y **B** en forma descendente, genere el arreglo **C** ordenado en forma ascendente.
8. Construya un programa en C, que reciba del usuario una palabra cualquiera, y luego reciba un carácter, y que imprima por pantalla cada posición donde el carácter aparece en la palabra ingresada. Si el carácter no aparece, debe indicar al usuario tal situación.
9. Modifique el programa anterior para que su programa reconozca el carácter ingresado por el usuario independiente si lo digitó en mayúsculas o en minúsculas. Además, si el usuario no ingresa una letra, su programa debe indicarlo al usuario.
10. INVESTIGACIÓN:
- a. La función “**scanf**” tiene un problema para leer arreglos de caracteres (“%s” de *string*): Si uno de los caracteres es un espacio, “**scanf**” lee solo hasta el espacio. Investigue qué otra función puede utilizar para leer un arreglo de caracteres completo, hasta que el usuario digite “Enter”.