

Tarea #3. Lenguajes de programación.

NOTAS:

- A) Las aplicaciones deben usar exclusivamente los conceptos aprendidos.
- B) Se debe crear un repositorio con el nombre tarea3
- C) Se deben crear archivos fuente para cada uno de los puntos, con nombres:
punto1.cpp, punto2.cpp, punto3.cpp, ... etc.

Las siguientes aplicaciones **DEBEN** incluir el uso de funciones, control de flujo, punteros, indexación y arreglos fijos.

1. Traductor a código Morse.

Entendiendo que una variable de tipo **string** es un arreglo de caracteres. Escriba un programa que reciba una frase introducida por el usuario y que imprima a consola la traducción a código Morse. El código morse en consola, debe tener una separación de un espacio entre letras y el carácter “/” entre palabras.

A	.-	N	-.
B	-...	O	---
C	-.-.	P	..-.
D	-..	Q	--.
E	.	R	.-.
F	...-	S	...
G	--.	T	-
H	U	..-
I	..	V	...-
J	.-...	W	.-.
K	-.-	X	-.-.
L	.-..	Y	-.--
M	--	Z	--..

Ejemplo: **HELLO WORLD** =-.. .-.. --- / .-- --- ._. .-. -.

- 2. Escribir un programa que genere un arreglo fijo de 100 números (con precisión de un solo decimal: ejemplo: -0.5, 0.5, 0.0 etc) aleatorios en el rango [1.0,- 1.0]. Escriba una función que calcule y retorne la desviación estándar. (Consultar la ecuación de desviación estándar).

3. Escribir una función que reciba un arreglo de 10 enteros y que retorne **un puntero** al valor máximo del arreglo. Escribir otra función que retorne **un puntero** al valor mínimo. En caso de que hallan múltiples valores máximos o mínimos, se debe retornar el puntero a cualquiera.
4. Empleando la función swap desarrollada en clase. Utilice esta función para crear otra función que refleje un arreglo de 10 enteros ingresado por el usuario. Ejemplo, si la entrada es el arreglo [1,2,3,4,5,6,7,8,9,10], la función debe modificar el mismo arreglo para producir [10,9,8,7,6,5,4,3,2,1].

5. Álgebra lineal.

Escriba una función que inicialice un arreglo bidimensional de tamaño 4x4, con valores enteros ingresados por el usuario.

Escriba una función que imprima un arreglo bidimensional de tamaño 4x4 de manera que se vea de la forma:

1	2	3	4
1	2	3	4
1	2	3	4
1	2	3	4

Escriba una función que reciba tres arreglos fijos cada uno 4x4; los dos primeros arreglos se consideran las entradas y el tercer arreglo la salida. Calcule el producto Hadamard (elemento a elemento) entre los primeros dos arreglos asignando el resultado al arreglo de salida.

Con ayuda de las funciones anteriores escriba una aplicación que pregunte al usuario por dos arreglos bidimensionales y que imprima a consola el producto Hadamard entre ellos.