



TRABAJO PRÁCTICO Nº2

Realizá cada ejercicio en lenguaje C siguiendo las indicaciones de la cátedra. En los casos en los que se requiera, desarrollá un planteo de solución (PS) y una prueba de escritorio (PE) antes de compilar y correr el programa.

Ejercicio 1. Escriba un programa que llene un arreglo con 10 números enteros aleatorios no repetidos entre 0 y 10 y lo imprima por pantalla.

Ejercicio 2. Escriba un programa que repita la operación del ejercicio anterior 10 veces y lleve el registro de la cantidad de veces que se repite cada uno de los números generados aleatoriamente. Al final, debe imprimir por pantalla la frecuencia de aparición de cada número.

Ejercicio 3. Escriba un programa que permita realizar la suma de dos vectores de dimensión 5 utilizando arreglos inicializados en el código (*hardcoded*) imprimiendo el resultado.

Ejercicio 4. Escriba un programa que llene un arreglo con 50 números enteros aleatorios entre 0 y 10, y calcule e imprima la moda, la media y la desviación estándar.

Ejercicio 5. Escriba un programa que dado un arreglo de 10 cadenas de caracteres *hardcoded* lo imprima al derecho (comenzando por la cadena de la primera posición) y al revés (comenzando por la última).

Ejercicio 6. Escriba un programa que a partir de un arreglo de 10 fechas (representadas como cadenas de caracteres) *hardcoded* indique la mayor y la menor.

Ejercicio 7. Escriba un programa que a partir de un arreglo inicialmente vacío de 10 cadenas de texto de longitud máxima 20 permita insertar una cadena nueva indicada por el usuario en orden alfabético. Al insertar una nueva cadena el programa debe desplazar las cadenas que ya se encuentren en el arreglo para hacerle lugar. Si el arreglo está lleno, se debe quitar la última cadena para hacer lugar a la nueva, a menos que la nueva sea la última por orden alfabético, en cuyo caso no se debe insertar. Ante cada inserción, el programa debe imprimir el arreglo en su nuevo estado. El programa termina al ingresar el usuario la palabra "fin". (PS)

Ejercicio 8. Escriba un programa que genere un arreglo de 20 números enteros aleatorios entre 0 y 9, lo imprima y luego elimine los duplicados, imprimiendo el arreglo resultante al finalizar.

Ejercicio 9. Escriba un programa que mantenga un arreglo de 20 cadenas de caracteres de longitud máxima 30 permitiendo al usuario ingresarlas desde el teclado solo si la cadena ingresada es al menos un 20% diferente a cualquier cadena ya ingresada. Para calcular la similitud entre cadenas, se deben comparar letra a letra y luego calcular (cantidad de letras iguales) dividido entre (tamaño de la cadena más larga). Luego de cada ingreso, se debe imprimir el arreglo con las cadenas ingresadas. (PS)

Ejercicio 10. Escriba un programa que contenga un arreglo de longitud 1000 de cadenas de texto de hasta 30 caracteres cada una. El programa debe permitir al usuario ingresar una oración y luego verificar si las palabras de la oración están presentes en el arreglo (diccionario) o no. Si una palabra no está presente, el programa debe indicarlo y darle la opción al usuario de agregarla. Al terminar de analizar el texto, el programa debe imprimir el estado actual del diccionario. El programa termina con la palabra "fin".

Ejercicio 11. Indique con sus palabras qué es lo que hace el siguiente programa:

```
#include <stdio.h>
```



```
int main() {
    int x[20] = {2, 5, 8, 12, 17, 20, 25, 30, 33, 40, 45, 50, 55, 60, 65,
70, 75, 80, 85, 90};
    int n = 20;
    int y[20];
    int m = 0;

    for (int i = 0; i < n; i++) {
        printf("%d ", x[i]);
        if (x[i] % 2 != 0) {
            y[m++] = x[i];
        }
    }
    printf("\n");

    for (int i = 0; i < m; i++) {
        printf("%d ", y[i]);
    }
    printf("\n");
    return 0;
}
```

Ejercicio 12. Reescriba el programa del punto anterior siguiendo buenas prácticas.

Ejercicio 13. Repita las consignas de los ejercicios 11 y 12 para el siguiente código:

```
#include <stdio.h>
int main() {
    int v[5];
    int e;

    for (int i = 0; i < 5; i++) {
        printf("Elemento %d: ", i + 1);
        scanf("%d", &v[i]);
    }

    scanf("%d", &e);

    for (int i = 0; i < 5; i++) {
        printf("%d ", v[i]);
    }
    printf("\n");

    for (int i = 0; i < 5; i++) {
        int p = v[i] * e;
        printf("%d ", p);
    }
    printf("\n");

    return 0;
}
```