

# Práctica 2

## Arreglos - Punteros - Strings

### Arreglos estáticos

1. Escriba un programa que genere 50 números enteros en forma pseudoaleatoria y que los almacene en un arreglo unidimensional. Luego de realizada la carga, informar la cantidad de números pares que se encuentran en posiciones impares del arreglo y también la cantidad de números impares que se encuentran en posiciones pares.
2. Escriba 3 funciones que reciban un arreglo unidimensional de float y que:
  - a. retorne el promedio de sus valores.
  - b. retorne el mínimo de sus valores.
  - c. retorne la posición del máximo de sus valores.

3. Sea  $A$  una matriz con  $m$  filas y  $n$  columnas y  $c$  un número. La multiplicación de una matriz  $A$  por un escalar  $c$  está dada por multiplicar cada entrada de  $A$  por  $c$ :

$$A'_{ij} = c \cdot A_{ij}$$

Por ejemplo:

$$2 \cdot \begin{bmatrix} 1 & 8 & -3 \\ 4 & -2 & 5 \end{bmatrix} = \begin{bmatrix} 2 \cdot 1 & 2 \cdot 8 & 2 \cdot -3 \\ 2 \cdot 4 & 2 \cdot -2 & 2 \cdot 5 \end{bmatrix} = \begin{bmatrix} 2 & 16 & -6 \\ 8 & -4 & 10 \end{bmatrix}$$

Escriba una función que reciba una matriz  $A$  y un número  $c$ , y que retorne en  $A$  la multiplicación escalar  $cA$ .

4. Sea  $A$  una matriz con  $m$  filas y  $n$  columnas. La matriz transpuesta, denotada con  $A^t$  está dada por:

$$A^t_{ij} = A_{ji}$$

En donde el elemento  $a_{ji}$  de la matriz original  $A$  se convertirá en el elemento  $a_{ij}$  de la matriz transpuesta  $A^t$ . Por ejemplo:

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}^t = \begin{bmatrix} a & c \\ b & d \end{bmatrix}$$

Escriba una función que reciba una matriz cuadrada  $A$  e imprima en pantalla su matriz transpuesta.

5. Dadas dos matrices  $A$  y  $B$  de  $m \times n$ , la suma  $(A+B)$  se calcula de la siguiente manera:

$$(A + B)_{ij} = A_{ij} + B_{ij}, \text{ donde } 1 \leq i \leq m \text{ y } 1 \leq j \leq n.$$

Por ejemplo:

$$\begin{bmatrix} 1 & 3 & 1 \\ 1 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 5 \\ 7 & 5 & 0 \end{bmatrix} = \begin{bmatrix} 1+0 & 3+0 & 1+5 \\ 1+7 & 0+5 & 0+0 \end{bmatrix} = \begin{bmatrix} 1 & 3 & 6 \\ 8 & 5 & 0 \end{bmatrix}$$

Escriba una función que reciba tres matrices  $A$ ,  $B$  y  $C$  de mismas dimensiones, y calcule la suma de  $A$  y  $B$  en  $C$ .

6. Escriba un programa que lea un número entero desde teclado y luego informe la cantidad de veces que aparece cada dígito en el mismo.
7. Se dispone de un arreglo con información de las precipitaciones de todos los días de todos los meses de los últimos 10 años. Se desea conocer para cada año el mes en que menos llovió y también cual es el año que tuvo menor precipitación promedio.  
*Nota: por simplicidad asuma que todos los meses tienen 30 días.*

## Punteros

- Rehaga el ejercicio 2 de la sección *Arreglos Estáticos*, pero en este caso escribiendo una única función que retorne los 3 valores pedidos.
- Complete la siguiente tabla con los valores correspondientes para cada variable de acuerdo con el código. Las primeras líneas ya fueron completadas a modo de ejemplo.

	a	b	c	p1	p2
int a = 1;	1	###	###	###	###
int b = 2;	1	2	###	###	###
int c = 4;	1	2	4	###	###
int * p1, * p2;	1	2	4	###	###
p1 = &a;	1	2	4	&a	###
p2 = &c;	1	2	4	&a	&c
*p1 = (*p2)++;					5
p2 = &b;					
*p1 += *p2;					
p1 = p2;					
++*p1;					
p1 = &a;					
*p2*=*p1;					
c = *p2+*p1;					
a = *p1 = ++*p2;					

*Nota: ### se utiliza para indicar que la variable no ha sido inicializada aún.*

- Analice y ejecute el siguiente código

```
include <stdio.h>

int main()
{
    int x[5]={1,2,3,4,5};
    int * p = x; pone el vector en p

    printf("La direccion memoria de p es %p \n", &p);
    printf("La direccion la que apunta p es %p \n", p);
    printf("El valor al que apunta p es %d \n", *p);

    p++;

    printf("La direccion memoria de p ahora es %p \n", &p);
    printf("La direccion la que apunta p ahora es %p \n", p);
    printf("El valor al que apunta p ahora es %d \n", *p);

    return 0;
}
```

- ¿Puede explicar qué significado tienen los valores impresos?
- ¿Qué ocurre cuando se incrementa en uno el puntero “p”?

4. Analice y ejecute el siguiente código:

```

1.  int main() {
2.      int vector[10]={10,20,30,40,50,60,70,80,90,100};
3.      int i;
4.      int *p= vector; a lo que apunta p le pone la direc del vector
5.      for (i=0; i<10; i++){
6.          *p += 3;
7.          printf("vector[%d] = %d \n", i, *p);
8.          p++;
9.      }
10.     return 0;
11. }
```

- ¿Qué imprime el programa?
  - ¿Por qué razón se ve modificada la variable *vector*? ¿Es correcto realizar esto?
  - ¿Qué ocurre si en la línea 7 cambia *\*p* por *vector[i]*? *imprime lo mismo*
  - Cambie la línea 8 por *p+=1*. ¿Qué sucede?
  - Cambie la línea 8 por *p+=2*. ¿Qué sucede?
5. Escriba una función que reciba un arreglo unidimensional e invierta el orden de sus valores. Por ejemplo:



*Nota: utilice la función swap definida en la Práctica 1.*

## Strings (Vectores de Caracteres)

- Escriba un programa que lea palabras desde teclado hasta leer la palabra "ZZZ". Informe la cantidad de palabras de longitud 5.
- Escriba un programa que lea palabras desde teclado hasta leer la palabra "XXX". Se requiere informar la cantidad de palabras ingresadas que terminan con la letra 'o'.
- Escriba una función que reciba un puntero a una cadena de caracteres y dos caracteres individuales, y que reemplace en la cadena todas las ocurrencias del primer carácter con el segundo. Además, la función debe retornar la cadena. ¿Retornar la cadena tiene alguna utilidad o no tiene sentido?
- La función *strlen* recibe un puntero a una cadena de caracteres y retorna su longitud (sin incluir el carácter nulo). Implemente su propia versión de la función *strlen*.
- La función *strcpy* recibe dos punteros a cadenas de caracteres y copia la cadena apuntada por el segundo (incluyendo el carácter nulo) en la cadena apuntada por el primero. Además, la función retorna el primer puntero. Implemente su propia versión de la función *strcpy*.  
*Nota: asuma que la primera cadena de caracteres tiene espacio suficiente para almacenar a la segunda.*
- Escriba un programa que lea una palabra desde teclado y luego informe la cantidad de veces que aparece cada letra de la palabra leída en ella misma.  
*Nota: asuma que la palabra está formada sólo por letras minúsculas.*
- Escriba una función que reciba un puntero a una secuencia de caracteres y que retorne 1 en caso de que la palabra contenida sea palíndroma o 0 en caso contrario.
  - Utilizando índices numéricos.
  - Utilizando punteros.

*Nota: una palabra palíndroma es aquella que se lee igual hacia izquierda a derecha que de derecha a izquierda. Por ejemplo: arenera, radar, reconocer, rotor, salas, seres, sometemos.*