

## Práctica 5 – Vectores

### Temas

- Estructura de datos: arreglos unidimensionales
- Arreglos paralelos
- Búsqueda – Inserción – Supresión

En los ejercicios en los que se describe un problema, implementar la solución en un programa Pascal eficaz, claro y sin pasos innecesarios.

Resolver en forma modular utilizando funciones y/o procedimientos según corresponda. Incluir el programa principal.

En todos los casos se deben armar lotes de prueba que contemplen todas las condiciones del algoritmo.

**Ej 1)** Dado un arreglo A de N elementos reales, desarrollar subprogramas para:

- a) Calcular la suma de sus elementos.
- b) Mostrar los elementos que se encuentran en posiciones pares.
- c) Calcular el máximo y el mínimo elemento.
- d) Generar un arreglo B con los elementos de A permutado ( $B[1]=A[N]$ ,  $B[2]=A[N-1]$  ...  $B[N]=A[1]$ ).

Escribir el programa principal con las invocaciones a cada uno de los subprogramas, mostrando los resultados correspondientes.

**Ej 2)** Dado un archivo de enteros, se pide almacenar en un vector V sólo los elementos impares y en otro vector W, los pares, ignorando si hubiese, los ceros.

**Ej 3)** Dado un arreglo V y un número K, cambiar todas las ocurrencias de K por 0.

**Ej 4)** Se pide compactar un arreglo (eliminar los ceros que contiene) generando otro arreglo.

**Ej 5)** Dado un archivo de enteros, almacenarlos en un arreglo, luego obtener el promedio de todos los números divisores del máximo y con ellos armar otro arreglo. Mostrar ambos arreglos y el promedio obtenido.

**Ej 6)** Escriba un programa que lea una frase (secuencia de caracteres separada por espacio, coma, punto) y mediante un menú de opciones calcule e informe:

- a) El número de palabras que contiene.
- b) La longitud de la palabra más larga.
- c) De todas las palabras, la que aparece antes en el diccionario. Analizar las alternativas de almacenar dicha palabra en un arreglo de caracteres o en un *string*.
- d) la frecuencia de cada letra (mayúscula o minúscula) que aparece en la frase.

No es necesario almacenar toda la frase en un arreglo.

**Ej 7)** Realizar la suma de dos números positivos muy grandes (de 10 cifras, por ejemplo). Implementar el programa principal que lea ambos números, invoque el subprograma que resuelve la suma y muestre el resultado.

**Ej 8)** Dado un arreglo V de N elementos ordenados en forma ascendente con componentes repetidas, generar W con la frecuencia de los elementos de V. Mostrar W

Ejemplo:  $N=10$ ;  $V=(2,2,3,3,3,3,7,9,9,9) \rightarrow W=(2,4,1,3)$

**Ej 9)** Dado un archivo de números enteros no nulos, almacenar en un arreglo A aquellos que formen una secuencia ascendente. A partir de A generar B con la misma cantidad de elementos de A pero poniendo ceros en aquellas componentes simétricas donde la simétrica derecha no sea múltiplo de la izquierda. Escribir ambos arreglos.

Ejemplo: Archivo: 5, 7, 1, 12, 15, -10, 10, 24, -25, 26, 50, 13

$A = (5, 7, 12, 15, 24, 26, 50) \rightarrow B = (5, 0, 12, 15, 24, 0, 50)$

**Ej 10)** Dado un arreglo A, generar dos arreglos paralelos B y C donde B contiene la suma de los elementos de A que están entre dos negativos y C la cantidad que se sumó. Mostrar B y C.

Ejemplo:

A=(3, 5, -1, 3, 8, 2, -5, 2, 10, -4, -8, 4, 6, 7, -3, 11, 2) → B=(13, 12, 0, 17) C=(3, 2, 0, 3)

**Ej 11)** De una competencia de atletismo se tienen los siguientes datos del lanzamiento de jabalina:

- Nombre del competidor (ordenado ascendentemente por este dato)
- 1º lanzamiento
- 2º lanzamiento

Se pide:

- a) Cuántos competidores superaron con el 2º lanzamiento el 1º.
- b) Nombre del atleta que registró la mejor marca.
- c) Dado el Nombre de un atleta, informar la marca que realizó en ambos lanzamientos.
- d) Generar un nuevo arreglo con los participantes que pasaron a la semifinal (aquellos que superaron una marca X establecida en el 1º ó 2º lanzamiento). Mostrar ambos arreglos.

**Ej 12)** En un archivo de texto se ha grabado por cada línea, una cadena (es una clave encriptada que puede contener letras, números y caracteres especiales (&, \$, #, etc.), y que siempre termina en letra), luego uno o más blancos y tres números separados por blancos, por último uno o más blancos y una letra.

Ejemplo: mU&1#hT34\*A 80 40 30 P

Se pide, a partir del archivo, cargar tres arreglos paralelos:

- *VClave* contendrá las claves sin los caracteres especiales, que registren en su encriptación hasta tres caracteres especiales (las claves que tengan más de tres invalidan la información de la línea, que no se almacena)
- *VProm* almacena el promedio de los tres números.
- *VCar* almacena el caracter que está al final de la línea, siempre que no coincida con la última letra de la clave, en cuyo caso invalida la información de la línea, que no se almacena)

Mostrar los arreglos generados, y luego para una letra y una clave que se ingresan por teclado:

- a) Para la letra ingresada, calcular e informar la clave de máximo promedio.
- b) Para la clave ingresada, informar el promedio y la letra asociada (puede no existir)

Resultados: si el archivo fuera

mU&1#hT34*A	80	40	30	P	
AsD4=H23**9+T	4	78	56	L	(1)
jYC%MkS6M	10	70	3	M	(2)
MnDe3!ku#z	6	8	19	A	



<i>VClave</i>	<i>VProm</i>	<i>VCar</i>
mU1hT34A	50	P
MnDe3kuz	11	A

(1) se descarta la línea porque la clave tiene más de 3 caracteres especiales.

(2) se descarta la línea porque la última letra de la clave coincide con la última de la línea.

**Ej 13)** En una estación de peaje hay N cabinas por donde pasan varios vehículos. Desde un archivo se lee la siguiente información:

- Número de cabina
- Tipo de vehículo (1=auto, 2=moto, 3=camioneta, 4=camión, 5=colectivo)
- Hora (0..23)

Las tarifas son: moto = \$150, auto y camioneta = \$300, camión = \$470, colectivo = \$390.

Si la hora está entre las 10 y las 18 sufre un incremento del 10%.

Calcular e informar:

- Cuántos vehículos atendió cada una de las N cabinas.
- Cuál fue la que más importe recaudó.

**Ej 14)** Una empresa de autos de alquiler grabó en cada línea del archivo la patente del vehículo (cadena de 7) y el total recaudado en el día. Se pide leer los datos en dos arreglos paralelos e informar:

- El promedio recaudado por auto.
- Las patentes de los vehículos que superaron en la recaudación X pesos (X es dato).
- La patente del que recaudó menos.

**Ej 15)** Una planta de silos registra el ingreso de camiones de sus clientes con cereal, por cada uno se tiene:

- Cliente
- Patente del camión
- Tipo de cereal ( 1-trigo, 2-girasol, 3-soja, 4-maiz )
- Toneladas

Se pide que lea los datos desde un archivo y mediante un menú de opciones iterativo calcule e informe:

- Porcentaje de toneladas de cada cereal respecto del total.
- Dada una patente, si es que existe, decir qué tipo de cereal transporta, las toneladas y a qué cliente pertenece.
- Generar un arreglo con los clientes que transportaron más de X toneladas, ordenado alfabéticamente.

Generar un archivo de texto, con un mínimo de 10 líneas, que obtenga los siguientes resultados:

- Trigo: 30%, Girasol: 20%, Soja: 40%, Maiz: 10%.
- Para la patente AB123CD -> Soja, 45, MDP3
- X = 30 -> [BAL1, MDP3, NEC3, TAN2]

**Ej 16)** Se pide compactar un arreglo, como en el ejercicio 4, pero sin usar estructura auxiliar.

**Ej 17)** Un médico tiene en un archivo de texto información sobre sus pacientes ordenado por su número de historia clínica. En cada línea:

- Número Historia Clínica (entero positivo, orden ascendente)
- Apellido y Nombre del Paciente (cadena de 30)

Se desea obtener un listado de los pacientes con apellidos que comienzan entre 'A' y 'F' ordenados alfabéticamente.