

Trabajo Práctico N° 4

Arreglos (unidimensional y bidimensional), Tipo Registro y Arreglos de Registros

Objetivo:

Incorporar a la estrategia de solución de problemas planteados herramientas que permitan manejar colecciones de datos similares, tipos simples y compuestos. Las estructuras son accedidas por medio de índices a posiciones predefinidas, las más utilizadas son los unidimensionales (vectores) y bidimensionales (matrices). Interiorizarnos en la creación de tipos definidos por el desarrollador (compuestos por más de un tipo simple), los registros nos permite tener la posibilidad de manipular más información dentro de una estructura de arreglo. Aplicar los conocimientos para el manejo de tipos registros. Aplicar los métodos de ordenación campos simples, registro y combinando campos de registro.

En Clase: Proponer una estrategia de abordaje a la resolución de los problemas planteados

1. En una fábrica de llantas se confeccionan entre 500 y 1000 llantas, por cada uno de los 4 turnos de 6 horas cada uno, con un tamaño de 14 a 19. Al terminar la fabricación de cada llanta, se la ubica en la pila del tamaño que le corresponde, pero si en el proceso de pulido se raya, dicha llanta se separa en otra pila de mercadería con detalles, también de acuerdo a su tamaño. Cuando una pila alcanza las 10 llantas (de las buenas o de las rayadas) se las embala en cajas y se las despacha al sector de producción.
Al inicio del día no existen llantas para ser embaladas, pero al finalizar un turno e iniciar un nuevo en el mismo día, se aprovecha el remanente de llantas apiladas listas para embalar.

Diseñar una solución que permita:

- a) Informar en cada turno, la cantidad total de cajas armadas por cada tamaño de llanta, indicando cuántas de ellas son con fallas y cuántas sin fallas, y ordenadas en forma decreciente según la cantidad de cajas de llantas sin fallas
- b) Calcular e informar el porcentaje de llantas sin fallas fabricadas en un día que no fueron embaladas (remanente final), con respecto a la producción diaria total, pero discriminado por cada tamaño

Con los conocimientos que dispone actualmente y desarrollando un programa en pseudocódigo se requiere disponer de la siguiente información

- a) Informar en cada turno, la cantidad total de cajas armadas por cada tamaño de llanta, indicando cuántas de ellas son con fallas y cuántas sin fallas.
 - b) Consolidar un informe que me permita visualizar al finalizar el proceso productivo el porcentaje de cajas embaladas con fallas y sin fallas sobre el total del empaque.
2. En un salón de fiestas donde están completas las 15 mesas de 8 sillas cada una, los pedidos de postres se atienden mediante una consola instalada en cada mesa. En cada consola se puede realizar de a un solo pedido cada 10 segundos, y la prioridad de atención se da por N° de mesa (de la 1 a la 15). En cada pedido se detalla entonces, el N° de la mesa, el N° de la silla, el código del postre preferido y el código del postre alternativo.

Al inicio de la cena se informa la cantidad de porciones que se prepararon de cada uno de los 20 postres diferentes que se sirven.

Cuando un invitado solicita su pedido, si hay en existencia se le sirve la porción de postre preferido, caso contrario se busca el postre alternativo. En caso de que el invitado haya solicitado su 3ra porción de un mismo postre, aparecerá en la consola el cartel 'Glotón' y no se servirá ningún postre.

Diseñar una solución que permita modelar la atención de los invitados, sabiendo que el servicio finaliza a los 20 minutos o bien cuando no existan más postres para servir.

- a) Informar por cada invitado de cada mesa, que al menos haya hecho un pedido de postre, el N° de la silla y mesa, total de postres diferentes que comió, total de avisos de 'Glotón' recibidos y cantidad de pedidos insatisfechos por falta de stock, ordenando este informe de menor a mayor por total de porciones comidas.
- b) Indicar además cuál fue el motivo del fin del servicio de atención.

Con los conocimientos que dispone actualmente y desarrollando un programa en pseudocódigo se requiere disponer de la siguiente información

- a) Informar por cada mesa, cantidad de invitados que comieron el postre preferido
- b) Informar por cada mesa, cantidad de invitados que comieron el postre alternativo
- c) Informar cuántos invitados fueron identificados como glotones

Ejemplo Unidimensional (Vectores): Se quiere calcular el promedio de un conjunto de 150 números enteros, los cuales son almacenados en un vector que inicialmente está lleno de ceros. Redondear el promedio a 2 decimales.

Análisis del Problema:

¿Qué Entradas se requieren (Tipo y Cantidad)? 150 números enteros.

¿Cuál es la Salida deseada (Tipo y Cantidad)? El valor promedio de los números leídos.

¿Qué método produce la Salida deseada? Debe construirse un vector lleno de ceros, cargarle los números a promediar, sumarlos y dividir por 150.

Diseño de la solución (por refinamientos sucesivos)

Primer intento: a grandes rasgos

Inicio

- Recorrer el vector cargando 0 en cada componente
- Recorrer el vector cargando en cada elemento un número entero obtenido de teclado
- Sumar los elementos del vector
- Dividir la suma obtenida por 150
- Informar el resultado de la división

Fin

Segundo intento: Desarrollo de la solución con Pseudocódigo.

Programa En Pseudocódigo:

Programa Promedio

```
// Programa escrito por: Gustavo
// Fecha: 20 de Junio de 2011
// Versión: 04
// Nombre del archivo: Promediar_Vector
// Este programa calcula el promedio de los 150 elementos de un vector.
```

Constantes

Valor_Inicial = 0

Tipos Estructurados

Vector = ARREGLO (150): Entero 3

Variables // del programa principal

Promedio: Real (4,2)

V: Vector

i: Entero 2

S: Real (5,2)

Hacer // Comienzo del programa

Imprimir: 'Hallar el promedio de 150 números enteros'

//Inicializar Vector

Repetir Para i:= 1, 150, 1

V(i):= Valor_Inicial

Fin Repetir Para

//Cargar Vector

Repetir Para i:= 1, 150, 1

Leer: V (i)

Fin Repetir Para

Fin Hacer

// Sumar

S: = 0

Repetir Para i: = 1, 150, 1

S: = S + V (i)

Fin Repetir Para

Promedio:= Redondear (S / 150, 2)

Imprimir: 'El promedio es ', Promedio

Fin Hacer

Fin Programa

En cada uno de los ejercicios que se plantean a continuación, determinar y explicitar las entradas necesarias, las salidas deseadas y el proceso que las relaciona ANTES de codificar.

ACLARACIÓN: Suele llamarse *tamaño* de un vector a la cantidad de elementos que contiene.

Ejemplo bidimensional (Matrices): Se desea saber si la suma de todas las posiciones de la Matriz M1 es igual a la suma de todas las posiciones de la Matriz M2. Se sabe que ambas son de 8 × 10 y contienen números reales en el intervalo [1,100] considerados con 2 dígitos decimales. Imprimir un mensaje apropiado que informe la situación.

Análisis del Problema:

¿Qué Entradas se requieren (Tipo y Cantidad)? Dos matrices conteniendo números reales

¿Cuál es la Salida deseada (Tipo y Cantidad)? Uno de los mensajes: “Las sumas coinciden.” o “Las sumas no coinciden.”, según corresponda.

¿Qué método produce la Salida deseada? Se suman todos los elementos de cada matriz y se comparan los resultados.

Diseño de la solución (por refinamientos sucesivos)

Primer intento: a grandes rasgos

Inicio

Recorrer la matriz M1 cargando en cada elemento un número real obtenido de teclado

Recorrer la matriz M2 cargando en cada elemento un número real obtenido de teclado

Sumar los elementos de la matriz M1

Sumar los elementos de la matriz M2

Se comparan ambas sumas y, en caso de ser iguales, informar “Las sumas coinciden”;
de lo contrario, informar “Las sumas no coinciden”;

Fin

Segundo intento: Desarrollo de la solución con Pseudocódigo.

Programa En Pseudocódigo:

Programa Primeras_Matrices

// Programa escrito por: Desarrollado por Gustavo

// Fecha: 20 de Junio de 2011

// Versión: 04

// Nombre del archivo: Primeras_Matrices

// Este programa compara la suma de los elementos de dos matrices.

Constantes

Cant_Filas = 8

Cant_Columnas = 10

Tipos Estructurados

Matriz = ARREGLO (Cant_Filas, Cant_Columnas): Real 5, 2

Variables // del programa principal

S1, S2: Real 6, 2

M1, M2: Matriz

C, F: Entero 2

Hacer // Comienzo del programa

Imprimir: ' Determinar si las sumas de los elementos de 2 matrices coinciden.'

//Cargar (M1-Matriz)

Repetir Para F:= 1, Cant_Filas, 1

Repetir Para C:= 1, Cant_ Columnas, 1

Leer: M1 (f, c)

Fin Repetir Para

Fin Repetir Para

//Cargar (M2)

Repetir Para F:= 1, Cant_Filas, 1

Repetir Para C:= 1, Cant_ Columnas, 1

Leer: M2 (F, C)

Fin Repetir Para

Fin Repetir Para

// Suma de los contenidos de las Matrices M1 y M2

Repetir Para F:= 1, Cant_Filas, 1

Repetir Para C:= 1, Cant_ Columnas, 1

S1:= S1 + M1 (F, C)

S2:= S2 + M2 (F, C)

Fin Repetir Para

Fin Repetir Para

// Determinar si las sumas S1 y S2 son iguales

Si (S1=S2) Entonces Imprimir: "Las sumas coinciden"

Sino Imprimir: "Las sumas no coinciden"

Fin Si

Fin Hacer

Fin Programa

En cada uno de los ejercicios que se plantean a continuación, determinar y explicitar las entradas necesarias, las salidas deseadas y el proceso que las relaciona **ANTES** de codificar.

Arreglos con Registro

Ejemplo: Se requiere un programa que permita cargar los datos de 230 empleados de una Cía. La información consiste en: nombre, apellido, sueldo básico, sucursal y área en que trabaja. Se tiene también una lista de sucursales y otra de áreas o sectores, las cuales se cargan al inicio para poder elegir entre ellas al introducir los datos de los empleados. ***Dejando a los alumnos aplicar los métodos de ordenamiento para generar un listado ordenado por :***

- ***En general por Apellido***
- ***Ordenado por Sucursal y Apellido***

Programa En Pseudocódigo:

Programa Mi_Primer_Registro_Arreglos

```
// Programador: G. Cerveri
// Fecha: 02 de Septiembre de 2010
// Versión: 03
// Nombre del archivo: Registros_Vectores
// Revisado por: Por Gustavo
// Fecha: 10 de octubre de 2008
```

Tipos Estructurados

Área = REGISTRO /* El Área de Trabajo o Sector */
Cod_Area: entero 2
Nombre_Area : carácter (10)
FIN REGISTRO

Lugar = REGISTRO /* Las Sucursales */
Cod_Lugar: entero 2
Sucursal : carácter (30)
FIN REGISTRO

Empleado = REGISTRO /* Los Empleados */
Legajo: entero 10
Apellido: carácter (30)
Nombre: carácter (30)
Sector: entero 2
Sucursal: entero 2
Sueldo: Real 6,2
FIN REGISTRO

Sectores = ARREGLO (10): Área /* Contiene las distintas secciones de la Cía */
Sucursales = ARREGLO (20): Lugar /*Contiene las distintas sucursales de la Cía */
Personal = ARREGLO (230): Empleado /* Contiene el Personal de la Cía */

Procedimiento Carga_Sectores (REF Sec_Cia : Sectores, Tot_Sec : Entero 2)

Variables

i : Entero 2

Hacer

Imprimir: ' Ingrese una detras de otra las Secciones de la Cía. '

Repetir Para i := 1, Tot_Sec, 1
Imprimir: ' Ingrese el nombre de la Sección Nro ' , i
Sec_Cia (i).Cod_Area := i
Leer: Sec_Cia (i).Nombre_Area
Fin Repetir Para

Fin Hacer

Fin Procedimiento

Procedimiento Carga_Sucursales (REF Suc_Cia : Sucursales, Tot_Suc : Entero 2)

Variables

i : Entero 2

Hacer

Imprimir: ' Ingrese una detras de otra las Sucursales de la Cía. '

Repetir Para i := 1, Tot_Suc, 1
Imprimir: ' Ingrese el nombre de la Sucursal Nro ' , i
Suc_Cia (i).Cod_Lugar := i
Leer: Suc_Cia (i).Sucursal
Fin Repetir Para

Fin Hacer

Fin Procedimiento

Procedimiento Carga_Empleados (REF Emp_Cia : Personal ; Tot_Emp : Entero 3 ; Sucur : Sucursales ; Secc : Sectores)

Variables

i : Entero 3

Hacer

Imprimir: ' Ingrese uno tras otro los Empleados de la Cía. '

Repetir Para i := 1, Tot_Emp, 1
Imprimir: ' Ingrese los datos del Empleado Legajo Nro ' , i
Imprimir: ' Ingrese el Nombre'
Leer: Emp_Cia (i).Nombre
Imprimir: ' Ingrese el Apellido '
Leer: Emp_Cia (i).Apellido
Imprimir: ' Sector al que pertenece: '
Emp_Cia (i).Sector := Elige_Sector (Secc)
Imprimir: ' Sucursal a la que pertenece: '
Emp_Cia (i).Sucursal := Elige_Sucursal (Sucur)
Imprimir: ' Ingrese Sueldo Basico '
Leer: Emp_Cia (i).Sueldo
Fin Repetir Para

Fin Hacer

Fin Procedimiento

Funcion Elige_Sector (Sec_Cia : Sectores) : entero 2

Variables

Op, i : Entero 2

Hacer

Imprimir: ' Lista de Sectores de la Cía. '

Repetir Para i := 1 , 10 , 1

Imprimir: ' # ', Sec_Cia (i).Cod_Area, ' Sector de ', Sec_Cia (i).Nombre_Area

Fin Repetir Para

Imprimir: ' Indique el Nro de Sector al cual Pertenece: '

Leer: Op

Retorno: Op

Fin Hacer

Fin Función

Función Elige_ Sucursal (Suc_Cia : Sucursales) : entero 2

Variables

Op, i : Entero 2

Hacer

Imprimir: ' Lista de Sucursales de la Cía. '

Repetir Para i := 1 , 20 , 1

Imprimir: ' # ', Suc_Cia (i).Cod_Lugar, ' Sector de ', Suc_Cia (i). Sucursal

Fin Repetir Para

Imprimir: ' Indique el Nro de Sucursal al cual Pertenece: '

Leer: Op

Retorno: Op

Fin Hacer

Fin Función

Variables // definición de las variables del programa principal

Emp : Personal // se declara una variable tipo Vector con Registro

S: Sucursales // se declara una variable tipo Vector con Registro

A: Sectores // se declara una variable tipo Vector con Registro

Hacer // Comienzo del programa

// Solo se requiere llamar procedimientos de carga

Carga_Sucursales (S, 20)

Carga_Sectores (A, 10)

Carga_Empleados (Emp, 230, S , A)

Fin Hacer

Fin Programa

IMPORTANTE

Lenguaje C

Los ejercicios que se han planteado en la **TP04** se pueden ir ejercitando en el Lenguaje C.

En el **CVG** en el área de **documentación** podrán acceder al **Manual de C** y al **Resumen C para alumnos**.

Los ambientes pueden ser el **DevC++** o cualquier **editor que compile C**. Hay herramientas en línea.

PSEUDOCODIGO

Se podrá acceder en el **CVG** en el área de **documentación** teoría y **PS** (un resumen de la sintaxis básica)

Los ejercicios de **Vectores** sugeridos

1. Implementar los programas en pseudocódigo adecuados para realizar las siguientes operaciones sobre un vector. En todos los casos hacer 2 versiones, una para un vector de enteros y otra para un vector de cadenas de caracteres (las comparaciones entre cadenas se realizan siguiendo el orden alfabético).
 - a) Buscar el máximo valor.
 - b) Determinar si un valor dado es o no un elemento del vector.
 - c) Determinar el elemento que ocupa la posición central (si el tamaño es impar) o el menor entre los dos centrales (si el tamaño es par).
2. Se cargan 2 vectores, uno con enteros pares y el otro con impares. Los ceros no se tienen en cuenta. Se quiere saber:
 - a) Cuántas veces el usuario ingresó un cero.
 - b)Cuál fue el vector que primero se terminó de cargar.
3. Un restaurante desea llevar el control de lo consumido en cada una de sus 50 mesas. La atención se realiza de la siguiente manera: al llegar gente al local se le asigna una mesa; por cada comensal se cobra un “cubierto” o servicio de mesa y se agrega el monto correspondiente a cada pedido. Al retirarse del local, se le cobra el monto acumulado. Debe considerarse la situación en que podría no haber mesas libres.

Los ejercicios sugeridos de **Matrices** para trabajar

1. Implementar el pseudocódigo adecuados para realizar las siguientes operaciones sobre una matriz.
 - a) Buscar el mínimo valor.
 - b) Determinar si un valor dado es o no un elemento de la matriz.
 - c) Determinar la posición ocupada por un valor dado en la matriz (si dicho valor no está en la matriz se devuelve 0).
 - d) Mostrar el contenido de la matriz recorrida de izquierda a derecha y de arriba a abajo.
 - e) Mostrar el contenido de la matriz recorrida de arriba a abajo y de izquierda a derecha.
2. Se desea cargar 2 matrices con números enteros y hacer las siguientes operaciones:
 - a) Multiplicar los valores de las dos matrices.
 - b) Sumar posición por posición de las dos matrices y colocarlos en un matriz resultado.
 - c) Comparar posición por posición los valores de las dos matrices y colocar en un matriz resultado el menor de ellos, en la misma posición.
3. Se cargan 2 matrices con números reales para realizar las siguientes operaciones:
 - a) Cargar en un vector la suma de cada columna de la primera matriz.
 - b) Cargar un vector con el promedio de cada fila de la segunda matriz.
 - c) Calcular el promedio de los valores de ambas matrices.
4. Se desea simular una batalla naval que cuenta con los siguientes barcos para cada jugador: 3 de una casilla, 2 de dos casillas y 1 de cuatro casillas. Los barcos pueden disponerse en horizontal o en vertical. Una vez ubicados los barcos, se ingresan las coordenadas del disparo y se emite un mensaje con la situación resultante del disparo (“AGUA”, o “IMPACTO” o “HUNDIMIENTO”). El juego termina cuando un jugador tiene todos sus barcos hundidos, siendo el otro jugador el ganador.

Los ejercicios sugeridos de **Tipo Registro con arreglos** para trabajar

Ejercicios sugeridos para trabajar en clase.

1. Un hotel tiene 100 habitaciones de distintas comodidades, la información la manejan con la siguientes registro:

Nro. Habitación, piso, tipo, precioXdia, confort, estado

Se quiere realizar las siguientes operaciones:

- a) Buscar habitaciones disponibles y listar sus datos
 - b) Imprimir el estado de las habitaciones, ordenadas por tipo
 - c) Liquidar una habitación, ingresando la cantidad de días que estuvo ocupada
2. Se tiene un arreglo cargado con datos de 5000 transacciones de una cartera de clientes de un Banco que usan la tarjeta de crédito TC. la información la manejan con la siguientes registro:

Nro. Cuenta, Fecha de operación, Monto, Nro de Comerciante.

Se quiere:

- a) Ordenarlo por Nro de Cuenta; si hay más de una transacción para cierto número de cuenta, el orden se decide por la fecha; si esto no alcanza, considerar al comerciante
 - b) Cantidad de clientes que realizaron estas transacciones
 - c) Realizar un listado de clientes con los montos totales de sus transacciones
3. Un comercio tiene el inventario de productos en una base de datos que contiene algo menos de 8000 ítems. Los registros de la base de datos tienen el siguiente diseño:

Nro. Ítem, Producto, Marca, Tipo, Costo, Stock Mínimo, Stock Actual

Se quiere realizar las siguientes operaciones:

- a) Insertar nuevos productos
- b) Borrar productos
- c) Modificar productos
- d) Buscar un producto

Simular la base de datos con un arreglo de 10000 elementos, de modo que siempre se encuentren ordenados por Nro. Ítem y que las últimas posiciones se encuentren disponibles. Si se alcanzara la capacidad máxima del vector, debe indicarse con un mensaje.

Ejercicios sugeridos para trabajar fuera del horario de clase.

4. Para realizar el control de tránsito una ciudad cuenta con 20 inspectores, cada uno de los cuales posee un dispositivo de almacenamiento manual con una capacidad de 2000 actas en el que se registran los siguientes datos:

nroActa, dominio, tipoInfraccion, fecha, hora, idInspector

Al regresar a la base de operaciones los inspectores entregan sus dispositivos a un operador que realiza las siguientes operaciones:

- Descarga todas las actas a un soporte único y las ordena por tipo de infracción y por inspector
- Visualiza un listado por tipo de infracción e inspector, indicando para cada tipo de infracción la cantidad total que hay y para cada inspector qué porcentaje de infracciones de ese tipo le corresponden. Además, al final debe indicarse el total de infracciones realizadas (de todos los tipos).

El listado debe realizarse de dos maneras:

- mostrando todos los datos, infracción por infracción
- Con el siguiente formato:

Infracción tipo 1	Inspector 1 porcentaje
	Inspector 2 porcentaje
	Inspector 3 porcentaje
...	...
	Total infracciones tipo 1 cantidad
...	...
Infracción tipo J	Inspector 1 porcentaje
	Inspector 2 porcentaje
	Inspector 3 porcentaje
...	...
	Total infracciones tipo n cantidad
...	...
	...
Cantidad total de infracciones cantidad	