

Diseño de algoritmos Registros

2.1.- Registros:

Definición, declaración y formas de representación.

Registro: Es un conjunto de datos elementales de diferentes tipos, lógicamente relacionados, los cuales se agrupan bajo un nombre común. Cada elemento del registro recibe el nombre de miembro ó campo del registro.

Ejemplo 1: Un registro de los pasajeros de un avión puede contener:

	Nombre	N Vuelo	Fecha	N Asiento	Costo Pasaje	Origen	Destino
Tipo Dato	car	ent	car	ent	real	car	car

Ejemplo 2: Un registro de los clientes y las transacciones de un banco

Registro de Clientes								
Nombre (car)	Cedula (ent)	Cuenta (car)	Fecha Nacimiento			Dirección		
			Dia (ent)	Mes (ent)	Año (ent)	Urb (car)	Calle (car)	Casa/Apto (car)
1								
:								
n								

Registro de Transacciones						
	Cuenta	Fecha Transaccion			Dep/Ret	Monto
		Dia	Mes	Año	Codigo	
	(car)	(ent)	(ent)	(ent)	(car)	(real)
1						
:						
n						

Definición de un registro :

Un registro es un tipo de dato diseñado por el usuario, que se debe definir antes de que se pueda utilizar.

Formato de la Definición :

Registro nombre_registro
tipo nombre campo_1
tipo nombre campo_2
:

Fin Registro

Ej :

Registro t_empleado
caracter nombre[10]
real sueldo
entero edad

Fin Registro

Variables tipo registro :

Al igual que a otro tipo de datos, a un registro se accede utilizando una variable ó variables, que se deben declarar después de la definición del registro, ya que la definición especifica simplemente el **nombre** y el **formato** del registro, pero no reserva almacenamiento en memoria.

Declaración de Variables tipo registro :

Cada declaración de variable para un registro dado, crea un área en memoria , donde los datos se almacenan de acuerdo al formato estructurado declarado.

Formato de la declaración :

Nombre_ Registro Nombre_variable

Ejemplo : t_empleado obrero, administrativo

En el ejemplo , se declaran las variables **obrero** y **administrativo** del tipo de registro **t_empleado** , por lo que cada una de las variables consta de los campos : nombre, sueldo y edad.

Acceso a los campos de un registro :

Se puede acceder a los campos de un registro usando el operador punto (.). Formato :
nombre_variable_ tipo_ registro. nombre_campo

Ejemplo :

obrero.nombre = Juan

Asigna al campo nombre de la variable obrero, el nombre Juan

Otra forma seria leyendo el nombre a través del teclado :

Leer (obrero.nombre)

♦ Campos simples:

En este caso los campos del registro están formados por variables de tipo simple, por ejemplo , un registro de la nota de un curso podría definirse de la manera siguiente:

```
registro t_alumno
    caracter   nombre[20]
    entero        cedula
    real        nota
fin_registro
```

♦ Campos constituidos por arreglos:

Los campos del registro pueden ser arreglos; por ejemplo, en el caso anterior , si se desea guardar cuatro notas por alumno se puede declarar el registro de la siguiente manera:

```
registro t_alumno
    car   nombre[20]
    ent   cedula
    real   nota[4]
fin_registro
```

nombre	cedula	nota[1]	nota[2]	nota[3]	nota[4]

Este registro permite almacenar las 4 notas de un alumno

Si se declara una variable **alumno** usando ese registro **t_alumno** :

Para hacer referencia a un elemento del registro, se escribe:

alumno.nota[1] = 15

alumno.nota[2] = 18

♦ Campos constituidos por registros:

En este caso, el registro interno debe definirse antes del registro principal.

Ejemplos:

1.-Se desea almacenar los datos de un producto en un registro. Cada producto debe guardar información concerniente a su código, nombre y precio. Diseñar la estructura de datos correspondiente y cargar los datos en el registro.

Algoritmo Creación de Registro

Inicio

// Definición del Registro

Registro t_producto

caracter codigo[3], nombre[10]

real precio

Fin Registro

// Declaración de la variable tipo registro

t_producto **detergente**

Mostrar (" Introduzca el código del producto ")

Leer (detergente.codigo)

Mostrar (" Introduzca el nombre del producto ")

Leer (detergente.nombre)

Mostrar (" Introduzca el precio del producto ")

Leer (detergente.precio)

Fin

2.- Una farmacia desea almacenar sus productos en una estructura de registros. Cada registro tiene los campos : codigo, nombre, precio, stock, fecha de vencimiento. La fecha debe guardar por separado día, mes y año , para lo cual se declarará otro registro.

Diseñar un algoritmo que permita :

- Diseñar la estructura de datos
- Mostrar los campos de un medicamento

Algoritmo Ejemplo

Inicio

Registro t_fecha

entero día, mes, año

Fin Registro

Registro t_medicina

carácter código[3], nombre[10]

entero precio, stock

t_fecha vencimiento

Fin Registro

t_medicina analgésico

Mostrar (analgésico.código)

Mostrar (analgésico.nombre)

Mostrar (analgésico.precio)

Mostrar (analgésico.stock)

Mostrar (analgésico.vencimiento.día)

Mostrar (analgésico.vencimiento.mes)

Mostrar (analgésico.vencimiento.año)

Fin

Arreglos de Registros :

Se puede crear un arreglo de registros tal como se crea cualquier tipo de arreglo, indicando el nombre y la dimensión. Los arreglos de registros son usados para almacenar datos en un archivo de disco.

Declaración de un Arreglo de Registros :

Formato :

nombre_registro nombre_arreglo [Dimensión]

Ejemplo:

t_medicina analgésico[100]

Acceso a los campos de un Arreglo de Registros :

Formato :

nombre_arreglo[posición].nombre_campo

Ejemplo :

Leer (analgésico[10].stock) //Accesa el campo stock del elemento ubicado en la posición 10 del arreglo de registros llamado analgésico.

Ejemplos :

1. Dada una tabla que contiene los datos NOMBRE, EDAD y PESO de un grupo de 300 estudiantes de una escuela primaria, Diseñar un algoritmo que permita cargar los datos y determinar cuantos estudiantes son mayores de 10 años y tienen un peso superior a 20 Kg.

algoritmo contar_alumnos

Inicio

//Definición de registro tipo **alumno**

Registro t_alumno

caracter nombre

entero edad

real peso

Fin Registro

//Declaración de Arreglo

t_alumno LISTA[50]

entero i, contador

//Cargar datos

repetir_desde (i = 1 ; i <= 50;i=i+1)

 mostrar("Introducir nombre, edad y peso del alumno", i)

 leer(LISTA(i).nombre)

 leer(LISTA(i).edad)

 leer(LISTA(i).peso)

fin_repetir_desde

contador = 0

Repetir desde i (i = 1 ; i <= 50;i=i+1)

Si ((LISTA[i].edad > 10) && (LISTA[i].peso > 20))

 contador = contador + 1

Fin Si

Fin Repetir desde

mostrar("Nº Alumnos EDAD > 10 y PESO > 20 : ", CONTADOR)

Fin

Aplicaciones sobre Arreglos de Registros

1.- Un restaurant posee una serie de insumos necesarios para la elaboración de los platos que ofrece en carta, por lo que es de vital importancia conocer cuando llegan a la existencia mínima permitida cada uno de sus 50 insumos, para lo cual se ejecutan los siguientes procesos :

- El encargado del almacén posee un listado con código, nombre, cantidad en existencia y stock mínimo
- El encargado de la cocina elabora solicitudes de insumos constantemente, en las cuales especifica : codigo y cantidad requerida.
- El encargado del almacén, al recibir la orden, envía la cantidad requerida y periódicamente, emite un listado con los códigos de los insumos que llegaron al stock mínimo.

Elabore un algoritmo que permita realizar las operaciones antes descritas, diseñando las estructuras de datos correspondientes. Usar un menú de opciones.

Análisis:

Diseño de las estructuras de datos :

El almacén lleva la información según la siguiente tabla:

	t_almacén			
	codigo	nombre	cant_ex	stock_min
1				
:				
50				

El pedido y el reporte tienen las siguientes estructuras:

	t_cocina	
	codigo	Cant_req
1		
:		
k		

	reporte
	codigo
1	
:	
m	

Diseño del algoritmo :

algoritmo Proceso de Restaurant
--

Inicio

// Declaracion de variables

Entero i, k, m, opcion = 1

Caracter reporte [50,10], respuesta = 's'

registro t_almacen

caracter codigo[10], nombre[20]

entero cant_ex, stock_min

fin registro

t_almacen almacen[50]

registro t_cocina

caracter codigo[10]

entero cant_req

fin_registro

t_cocina cocina[50]

repetir mientras (opcion<4)

 mostrar("Introduzca su opción :")

 mostrar("[1] Registrar datos")

 mostrar("[2] Elaborar Pedido")

 mostrar("[3] Enviar Pedido y Elaborar Reporte")

 mostrar("[4] Salir")

 leer(opcion)

en caso de (opcion)

caso 1: // Registrar datos

repetir_desde i = 1 **hasta** i==50

Mostrar (" Introduzca codigo,nombre, cantidad y
stock del insumo ", i)

leer(almacen[i].codigo)

leer(almacen[i].nombre)

leer(almacen[i].cant_ex)

leer(almacen[i].stock_min)

fin_repetir_desde

caso 2: // Elaborar Pedido

 k =0

repetir mientras(resp== 's')

```

        k = k+1
        mostrar("Introduzca codigo del insumo ", k)
        leer(cocina[k].codigo))
        mostrar("Introduzca cantidad del insumo ", k)
leer(cocina[k].cant_req)
        mostrar("Desea solicitar otro insumo s/n")
        leer(respuesta)

    fin repetir mientras
    mostrar ("Reporte de Insumos requeridos :")
    mostrar ( " codigo          cantidad requerida" )
    repetir desde i=1 hasta i== k
        mostrar ( cocina[i].codigo , cocina[i].cant.req )
    fin repetir desde

```

caso 3: // Enviar Pedido

```

    m = 0
    repetir_desde i = 1 hasta i==k
        repetir_desde j = 1 hasta j==50
            si(cocina[i].codigo = almacen[j].codigo)

                almacen[j].cant_ex=almacen[j].cant_ex –cocina[i].cant_req
                si(almacen[j].cant_ex <= lmacen[j].stock_min)

                    m = m+1
                    reporte[m] = cocina[i].codigo

            fin_si
        fin_si
    fin repetir_desde
fin_ repetir_desde

```

// Elaborar Reporte

```

    mostrar("Reporte de Componentes Faltantes")
    repetir_desde i = 1 hasta i == m
        mostrar(REPORTE[i])
    fin_repetir_desde

```

caso 4: // Finalizar

```

    Mostrar ( " Fin del programa " )

```

Fin en caso de

Fin repetir mientras

Fin

- 2.- .-Una empresa que se dedica a la venta de vehículos cuenta con 10 vendedores. La información sobre el número de vehículos vendidos es registrada semanalmente por cada uno de los vendedores, indicando además su nombre y cédula.
- Se desea diseñar un algoritmo que , asumiendo registrada la información por vendedor , permita :
- Dado el número de cédula, mostrar el total de vehículos vendidos en el mes
 - A efectos de pagar un bono , se desea conocer el nombre del vendedor con mayores ventas .

Diseño del Algoritmo :

Algoritmo Ventas de vendedores

Inicio // Declaración de variables

entero i, j, tot_mes[10], ced, mayor

registro t_vendedor

caracter nombre[10]

entero cedula, vent_sem[4]

fin registro

t_vendedor vendedor[10]

mostrar("Introduzca el número de cédula")

leer(ced) **//Cálculo ventas por mes para todos los vendedores**

repetir desde i=1 **hasta** i==10

tot_mes[i] = 0

repetir desde j=1 **hasta** j==4

tot_mes[i] = tot_mes[i] + vendedor[i] . vent_sem[j]

fin repetir desde

fin repetir desde

//Búsqueda de la cédula leída

repetir desde i=1 **hasta** i==10

si (vendedor[i] . cedula == ced)

mostrar(" Cantidad vendida en el mes =", tot_mes[i])

fin si

fin repetir desde

//Búsqueda del vendedor con mayor cantidad de ventas

mayor = tot_mes[1]

repetir desde i=1 **hasta** i==10

si (tot_mes[i] > mayor)

mayor = tot_mes[i]

pos = i

fin si

fin repetir desde

mostrar("vendedor con mayor cantidad de ventas = ", vendedor[pos] . nombre)

fin

EJERCICIOS PROPUESTOS

1.- Una librería posee 20 tipos de textos que son alquilados a diversos clientes. Los textos se tienen codificados, por lo que por cada texto se tiene : código, título, autor, cantidad existente.

Además, existe un registro de los 100 usuarios, con los siguientes datos : número de carnet, nombre, estado (A = activo, S = suspendido).

Cada uno de los 100 usuarios puede alquilar un texto solo si no está suspendido y para ello se debe registrar en el momento de alquilar un libro : número de carnet, código del libro, fecha de entrega programada.

Si al entregar el texto, la fecha de entrega es posterior a la fecha programada el día del alquiler, se coloca el estado como suspendido.

Se pide diseñar un algoritmo que permita ejecutar los siguientes procesos :

- Registrar los 20 textos y los 100 usuarios, diseñando las estructuras de datos correspondientes.
- Alquilar un texto a un cliente determinado.
- Entregar un texto por parte del cliente, considerando el caso de suspensión si la entrega es posterior a la fecha programada.

La ejecución de cada uno de los procesos debe ser seleccionada a través de un menú de opciones.

2.- Una empresa dedicada a la conexión de televisión por cable posee una cartera de 50 clientes, donde por cada cliente se tiene : cédula, dirección, teléfono, número de plan de conexión. La empresa ofrece 5 planes diferentes, donde por cada plan ofrecido se guarda : número de plan de conexión, listado de los 14 canales que lo conforman, costo del plan.

Diseñar un algoritmo para :

- Registrar la información de los 5 planes
- Registrar los 50 clientes y si es necesario, incluir nuevos clientes, sin excederse de 100.
- Modificar el costo de cualquier plan.
- Generar el recibo de pago de cualquier cliente.

3.- Una empresa dedicada a la venta de equipos electrónicos efectúa ventas al contado y a crédito. Cuando la venta se efectúa al contado se requiere emitir una factura con los siguientes datos : nombre del cliente, fecha de venta, cantidad de artículos, precio unitario y total a pagar. Si la venta se efectúa a crédito se requiere la siguiente información : nombre del cliente, teléfono, dirección, fecha de venta, deuda, cuota mensual, tasa de interés y estado (0 : deuda pendiente, 1 : deuda cancelada).

Se pide desarrollar un algoritmo que :

- Registre la información correspondiente a cada tipo de venta (crédito o contado).
- Registre el pago de abonos a la deuda
- Consulte la deuda de un cliente
- Muestre el listado de clientes permitiendo seleccionar : lista completa, lista de clientes con deuda.

4.- La información de un almacén se guarda en un registro con las siguientes características : código de los n artículos vendidos en el almacén y la existencia (número de unidades por cada artículo).

Por cada proveedor o cliente que llegue al almacén se graba un registro con los siguientes datos:

- tipo de transacción (1 para proveedor : recibo, 2 para cliente : venta)
- código del artículo recibido o vendido
- Número de unidades recibidas o vendidas

Se requiere manejar el proceso de cada transacción (recibo o venta) y en cada caso , mostrar el registro del almacén actualizado.

5.-Un sistema que permite administrar una librería usa una estructura que contiene el **código** del libro, **título** y el **precio** del libro. Diseñar un algoritmo que permita las siguientes opciones presentadas en forma de menú.

- (1) Registrar Libro
- (2) Ordenar por precio en forma ascendente/descendente
- (3) Buscar el libro de mayor/menor precio
- (4) Salir

El sistema deberá ser diseñado de manera que:

- ◆ Se debe poder registrar varios libros y finalizar el ingreso cuando se desee.
- ◆ El ordenamiento puede ser hecho de forma ascendente o descendente.
- ◆ La búsqueda puede ser del de mayor o el de menor precio.