

Estructuras de datos: Registros II

LISTAS: Array de Registros

Listas

4. arrays de estructuras de tipo Registro.

Cuando los elementos de un array son estructuras de tipo registro, el array recibe el nombre de array de estructura, matriz de registros o simplemente, Lista. Es una construcción muy útil para administrar datos que están en sintonía con objetos de la vida cotidiana.

4. a. Definir arrays de Registros.

Para definir "Un Vector de registros" primero hay que definir un registro de tipo estructura y luego definir en base a esta estructura, un array.

```
tipo
    registro regArticulo
        entero: codigo        //identificador articulo
        real: precio          //precio venta articulo
        entero: cantidad      //items de venta
    fin_registro

tipo
    array[1..20] de regArticulo: arrArticulo

var
    arrArticulo: articulo      //vector de artículos - 20 registros
```

Para acceder a los miembros de datos del primer registro del vector (base 1), la notación sería:

```
articulo[1].codigo ← 100
articulo[1].precio ← 43.50
articulo[1].cantidad ← 2
```

el 2do. sería:

```
articulo[1].codigo ← 200
articulo[1].precio ← 55.50
articulo[1].cantidad ← 3
```

etc.

IMPORTANTE: Como se puede deducir, siendo un array, debemos aplicar todas las operaciones fundamentales de los arrays, es decir Bucles para sus recorridos, ordenamiento, búsqueda, etc.



Estructuras:

En los compiladores las estructuras son conocidas como *registros*. Están formados por miembros de datos heterogéneos (campos).

Dependencia funcional:

Los Registros almacenan datos relacionados funcionalmente entre si. Así. Los datos de un cliente, es un registro que incluye: Código, Razon Social, RUC.

Proyecto. Hagalo Ud. Mismo.

Ejemplo comentado.

El siguiente ejemplo expone las características heredadas de los array en los vectores de registros.

Problema. Porcentaje de aprobados y desaprobados de una lista de alumnos (max 45 alumnos por aula) matriculados en la asignatura de Matematica Basica I.

Analisis.

La lista de alumnos y sus notas correspondientes es la fuente de datos a consultar iterativamente para determinar el numero de aprobados y desaprobados en tanto por cien. Estructura de datos:

Registro de alumnos

Apellido - cadena de 20 caracteres

Nota - nota final del curso

Lista de alumnos:

Array[1..45] de registro de alumnos

Las rutinas a implementar son:

- * Guardar registro alumno en disco
- * Leer desde disco lista de alumnos y almacenarla en un array de registros.
- * calcular el porcentaje de aprobados y desaprobados.

Especificaciones de E/S

Entrada: Lista de alumnos tipo Vector de registros (Apellido, nota)

Salida: porcentaje de aprobados y desaprobados.

Pseudocodigo.

Algoritmo. Porcentaje de aprobados-desaprobados

Const N = 45 //numero máximo por aula

tipo

registro: regAula
cadena: apellido
entero: nota

fin_registro

tipo

array[1..N] de regAlula: arrAula

var

regAula: alumno
arrAula: lista
entero: aprobados, desparobados

inicio

// menu de opciones

fin



```
/* Modulos: aquí todas las sub-rutinas que usa la aplicación
   Definición de prototipos
*/
//leerRegistro()

//escribirRegistro()

//escribirRegistroEnDisco()

//leerRegistrosDeDisco()

//calcularAprobadosDesaprobados()
```

OBSERVACION: Use este pseudocódigo, como sugerencias para, implementar una de las sub-rutinas del programa modular:

Convierta a Sub-rutina: calcular por ciento de aprobados-desaprobados

```
aprobados ← 0
desaprobados ← 0

//iterar lista para contar aprobados-desaprobados
Desde(i ← hasta N )hacer
    si(lista[i].nota <= 10)entonces
        desaprobados ← desaprobados + 1
    fin_si
    aprobados ← aprobados + 1
fin_desde
// calcular por ciento de aprobados y desaprobados
escribir('Aprobados  :', aprobados / N *100)
escribir('Desaprobados: ', desaprobados / N * 100)
```

5. Operaciones sobre arrays de tipo Registro.

5.a. Operaciones de Ordenamiento de registros.

5.b. Operaciones de Búsqueda de registros.