

Algoritmos y Programación Estructuras de Datos

Arrays Compuestos Registros

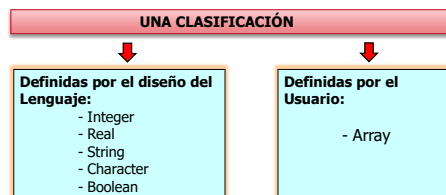
Año: 2018

Prof. Miguel A. Fernández

Algoritmos y Programación

Repasando:

Tipo de Dato: Atributo de un objeto de dato que especifica el rango de valores y las operaciones posibles.



Arreglos Compuestos - Registros

Repasando:

Arreglos Unidimensionales Simples:

- Se caracterizan por el procesamiento de repeticiones de un solo tipo de dato.

Ejemplo:

Para almacenar el nombre de los 80 alumnos de la materia Algoritmos:



Arreglos Compuestos - Registros

Cómo lo definíamos:

```

Program Prueba
Type
  Alumnos = array [1..80] of string;
Var
  Alumno : Alumnos;
  
```

IMPORTANTE

El tipo definido por el usuario (Alumnos) DEBE tomar como referencia un tipo definido por el diseño del lenguaje (TIPO PRIMITIVO).

Para imprimir el dato del elemento ubicado en la posición 77:

Imprimir Alumno(77)

Arreglos Compuestos - Registros

Arreglos Unidimensionales Simples Encadenados:

- Son varios arreglos que tienen la misma dimensión y que almacenan datos que tienen alguna relación entre sí.

Ejemplo:

Para almacenar el Nro.de matrícula y el nombre, de los 80 alumnos de Algoritmos:



Arreglos Compuestos - Registros

Cómo lo definíamos en Pascal:

```

Program Prueba
Type
  Matriculas = array [1..80] of integer;
  Nombres = array [1..80] of string;
Var
  Matrícula : Matriculas;
  Nombre : Nombres;
  
```

Para imprimir los datos que corresponden al alumnos ubicado en la posición 77:

Imprimir Matrícula(77), Nombre(77)

Arreglos Compuestos - Registros

Ejemplo:

Ingresar los datos de los 80 alumnos y luego de finalizado el ingreso codificar un algoritmo de consulta tal que ingresando el número de matrícula se imprima el nombre del alumno.

```

Program Prueba
Type
  Matriculas = array [1..80] of integer;
  Nombres = array [1..80] of string;
Var
  Matricula : Matriculas;
  Nombre : Nombres;
  i : integer;
  Mat : integer;
  Encontrado: Boolean;

```



Arreglos Compuestos - Registros

```

Program Prueba
Type
  Matriculas = array [1..80] of integer;
  Nombres = array [1..80] of string;
Var
  Matricula : Matriculas;
  Nombre : Nombres;
  i : integer;
  Mat : integer;
  Encontrado: Boolean;

```

```

Inicio
Para i = 1, 80, 1
  Ingresar Matricula(i)
  Ingresar Nombre(i)
FinPara

```



```

Ingresar MAT
Mientras MAT <> 0
  Encontrado := False
  i := 0
  Mientras i < 80 and Encontrado = False
    i := i + 1
    Si Matricula(i) = MAT
      Mostrar Nombre(i)
      Encontrado := True
    FinSi
  FinMientras
  Si Encontrado = False
    Mostrar "Matricula no Encontrada"
  FinSi
  Ingresar MAT
FinMientras
Fin

```

Arreglos Compuestos - Registros

OBJETO DE DATOS

Se refiere a un agrupamiento en tiempo de ejecución de uno o más datos en una computadora virtual.

RESUMIENDO:

¿ Qué almacenan los "objetos de datos" vistos hasta ahora ?

- Los objetos de datos simples o escalares pueden almacenar un solo dato (variables).
- Los objetos de datos tipo arreglo pueden almacenar conjuntos de datos pero del mismo tipo .

Arreglos Compuestos - Registros

TODO BIEN, PERO.....

- A menudo se necesita trabajar con varios datos de diversos tipos.
- En estos casos ni variables escalares ni arreglos son adecuados, ya que con ninguno de ellos es posible representar esta clase de objetos en forma natural.
- Podrían utilizarse arrays encadenados, pero a veces resulta muy engorroso su procesamiento.
- Para resolver este problema los lenguajes de programación:

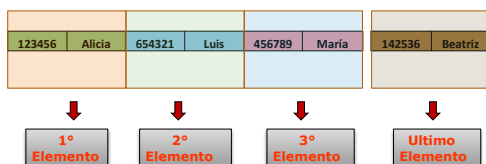
Proveen mecanismos que permiten agrupar datos incluso de distintos tipos en un solo array



Arreglos Compuestos - Registros

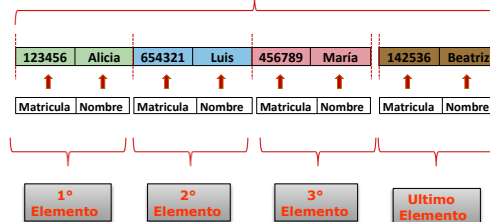
En el caso del ejemplo anterior, podríamos agrupar todos los datos en un solo arreglo.

El arreglo almacenará los dos datos de cada alumno (matrícula y nombre) en un mismo elemento.



Arreglos Compuestos - Registros

Vector "Alumno"



Cada elemento del vector "Alumno" tiene dos Campos:
- Matrícula
- Nombre



Arreglos Compuestos - Registros

Como se define: (En Pascal):

Pascal proporciona un tipo nuevo de datos llamado "Record" que permite definir estructuras que agrupan datos de distintos tipos.

Para operar con un arreglo unidimensional de nombre "alumno" que contenga 3 campos: Matrícula, Nombre y Teléfono; se define:

```

Type
  Registro = Record
    Matricula : Integer;
    Nombre : string;
    Telefono : Integer;
  end;
  Alumnos = array [ 1..80 ] of Registro;
Var
  Alumno : Alumnos;

```



Arreglos Compuestos - Registros

En Resumen:

Se ha definido:

- Una objeto de nombre: "Alumno"
- El objeto "Alumno" es del tipo "Alumnos"
- El tipo "Alumnos" es un array de 80 elementos donde cada elemento es del tipo "Registro"
- El tipo "Registro" es un Record compuesto por los siguientes campos:
 - "Matricula", del tipo primitivo entero
 - "Nombre", del tipo primitivo string
 - "Telefono", del tipo primitivo entero

```

Type
  Registro = Record
    Matricula : Integer;
    Nombre : String;
    Telefono : Integer;
  end;
  Alumnos = array [1..80] of Registro;
Var
  Alumno : Alumnos;

```



Arreglos Compuestos - Registros

Formas de referenciamiento:

a) Algunos lenguajes aceptan referenciamiento por nombre de campo solamente:

```

Imprimir Matricula(10)
Ingresar Nombre(10)

```

b) Pascal requiere la siguiente sintaxis:

Nombre Arreglo [subíndice] . Nombre de Campo

Ejemplo:

```

Imprimir Alumno[10].Matricula
Ingresar Alumno[10].Nombre

```



Registros

EJEMPLO

Ingresar los datos y luego, ingresando el N° de Matrícula Mostrar el nombre y el teléfono del Alumno.



```

Inicio
  Para i = 1,80,1
    Ingresar Alumno[i].Matricula
    Ingresar Alumno[i].Nombre
    Ingresar Alumno[i].Telefono
  FinPara
  Ingresar MAT
  Mientras MAT <> 0
    Encontrado := False
    i := 0
    Mientras i < 80 and Encontrado = False
      i := i + 1
      Si Alumno[i].Matricula = MAT
        Mostrar Alumno[i].Nombre
        Mostrar Alumno[i].Telefono
        Encontrado := True
      FinSi
    FinMientras
    Si Encontrado = False
      Mostrar "Matricula no Encontrada"
    FinSi
    Ingresar MAT
  FinMientras
Fin

```



Arreglos Compuestos - Registros

**¡ TIEMPO
DE
TRABAJO !**



Arreglos Compuestos - Registros

Se desea imprimir un listado con el detalle con datos de cada equipo que ha participado en el campeonato de fútbol organizado por el centro de Estudiantes de la Facultad.

De los 10 equipos que intervinieron se ingresa:

NOE - Nombre del equipo
CPG - Cantidad de Partidos Ganados
CPE - Cantidad de Partidos Empatados
CPP - Cantidad de Partidos Perdidos

Se requiere:

- Ingresar los datos y almacenarlos en una estructura utilizando el tipo registro.
- Codificar un módulo de consulta tal que ingresando el nombre del equipo se muestren los datos almacenados y la cantidad de puntos obtenidos por el equipo. [se asigna 3 puntos por partido ganado y 1 punto por partido empatado]
- Imprimir un listado que contenga:

Nombre Equipo	PARTIDOS			Puntos
	Ganados	Empatados	Perdidos	

Al finalizar el listado imprimir el Nombre del equipo que mas puntos obtuvo.



Program Campeonato

Type

Registro = Record

NOE : String [30];

PGA : Integer;

PEM : Integer;

PPE : Integer;

end;

Equipos = array [1..10] of Registro;

Var

i : Integer;

NOM : String;

ESTA : Boolean;

Equipo : Equipos;

Inicio

Ingresar i

Mientras i > 0 and i < 10

Ingresar Equipo[i].NOE, Equipo[i].PGA

Ingresar Equipo[i].PEM, Equipo[i].PPE

Ingresar i

FinMientras

Solución

Mostrar "Ingrese Nombre del Equipo: "

Ingresar NOM

Mientras NOM <> " "

i := 0

ESTA := False

Mientras i < 10 and ESTA = False

i := i + 1

Si Equipo[i].NOE = NOM

PUN := Equipo[i].PGA * 3 + Equipo[i].PEM

Mostrar Equipo[i].NOE

Mostrar Equipo[i].PGA

Mostrar Equipo[i].PEM

Mostrar Equipo[i].PPE

Mostrar PUN

ESTA := True

FinSi

FinMientras

Si ESTA = False

Mostrar "No existe Equipo"

FinSi

Ingresar NOM

FinMientras

FIN

Solución posible

MAX := Low_Value

Para i = 1, 10, 1

PUN := Equipo[i].PGA * 3 + Equipo[i].PEM

Imprimir: Equipo[i].NOE, Equipo[i].PGA

Imprimir: Equipo[i].PEM, Equipo[i].PPE

Imprimir: PUN

Si PUN > MAX

MAX := PUN

EQG := i

FinSi

FinPara

Imprimir: "Equipo con + Puntos: " Equipo[EQG].NOE

FIN

Arreglos Compuestos - Registros

Se desea procesar los datos de los alumnos que cursan las distintas carreras de la Facultad. Se debe codificar un programa que contemple:

1. Ingreso de Datos: De cada alumno Ingresar:
COMA – Código de matrícula del alumno
APEN – Apellido y Nombre
CMAT – Cantidad de Materias aprobadas
Código de carrera que cursa (rango de 1 a 10)
Todos estos datos se deben almacenar en un array de tipo registro.

2. Listado del padrón: Imprimir un listado de todos los alumnos que contenga:

Apellido y Nombre	Nombre de la carrera que cursa	Cantidad Materias aprobadas
-------------------	--------------------------------	-----------------------------

* Los nombres de cada carrera se ingresa al inicio del algoritmo

Al finalizar el listado imprimir:
Cantidad de alumnos discriminado por carrera.
Nombre de la carrera que tiene la mayor cantidad de alumnos.

Arreglos Compuestos - Registros

FIN

MUCHAS GRACIAS