

Programación I

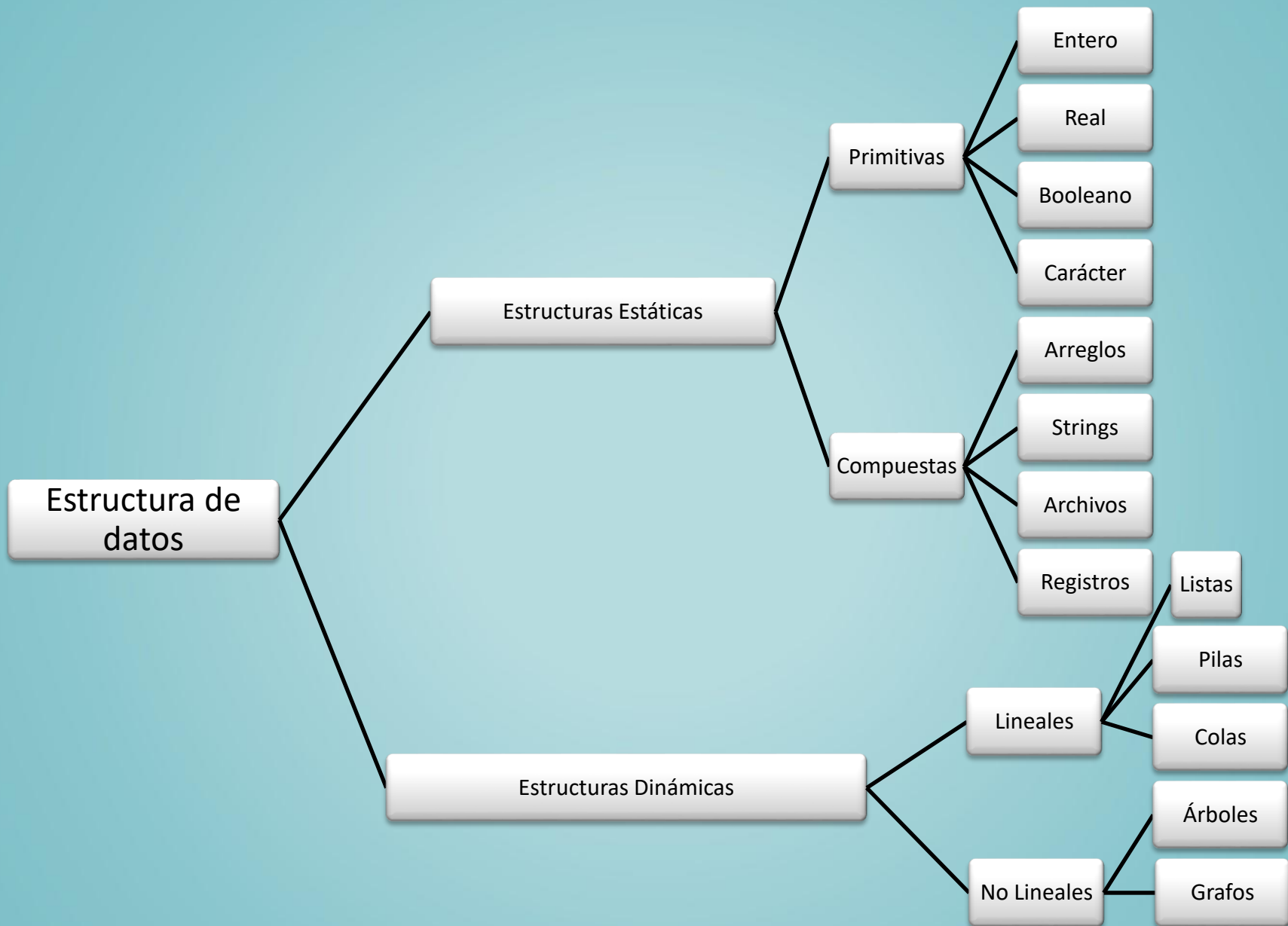
```
ataAssistance = new  
ataAssistance($nidBenefit, $subsidized, $systemBen  
centage, $sobs);  
return $odataAssistance;  
  
function recoverataAssistance($nidBenefit, $bank) {  
    $odataAssistanceBD = $this->startataAssistanceBD($bank)  
    $odataAssistance = $odataAssistanceBD->recover($nidBenefit)  
    return $odataAssistance;  
}  
  
function recoverataAssistance($bank) {  
    $odataAssistanceBD = $this->startataAssistanceBD($bank)  
    $object = array();  
    $object = $odataAssistanceBD->recoverAll();  
    return $object;  
}
```

UNIDAD 4

Manejo de Estructuras Secuenciales de Datos. Arreglos unidimensionales y multidimensionales. Operaciones con Arreglos: Recorrido, búsqueda, inserción. Cadena de Caracteres. Operaciones con cadenas. Concepto de Estructuras. Procedimientos de Ordenación y Búsqueda Binaria. Estructuras Dinámicas: Listas. Pilas y Colas. Árboles binarios. Operaciones con estructuras dinámicas: Recorrido, búsqueda, inserción. Árboles binarios. Archivos.

Estructuras de Datos

Es una colección de datos que se caracterizan por su organización y las operaciones que se definen en ella.

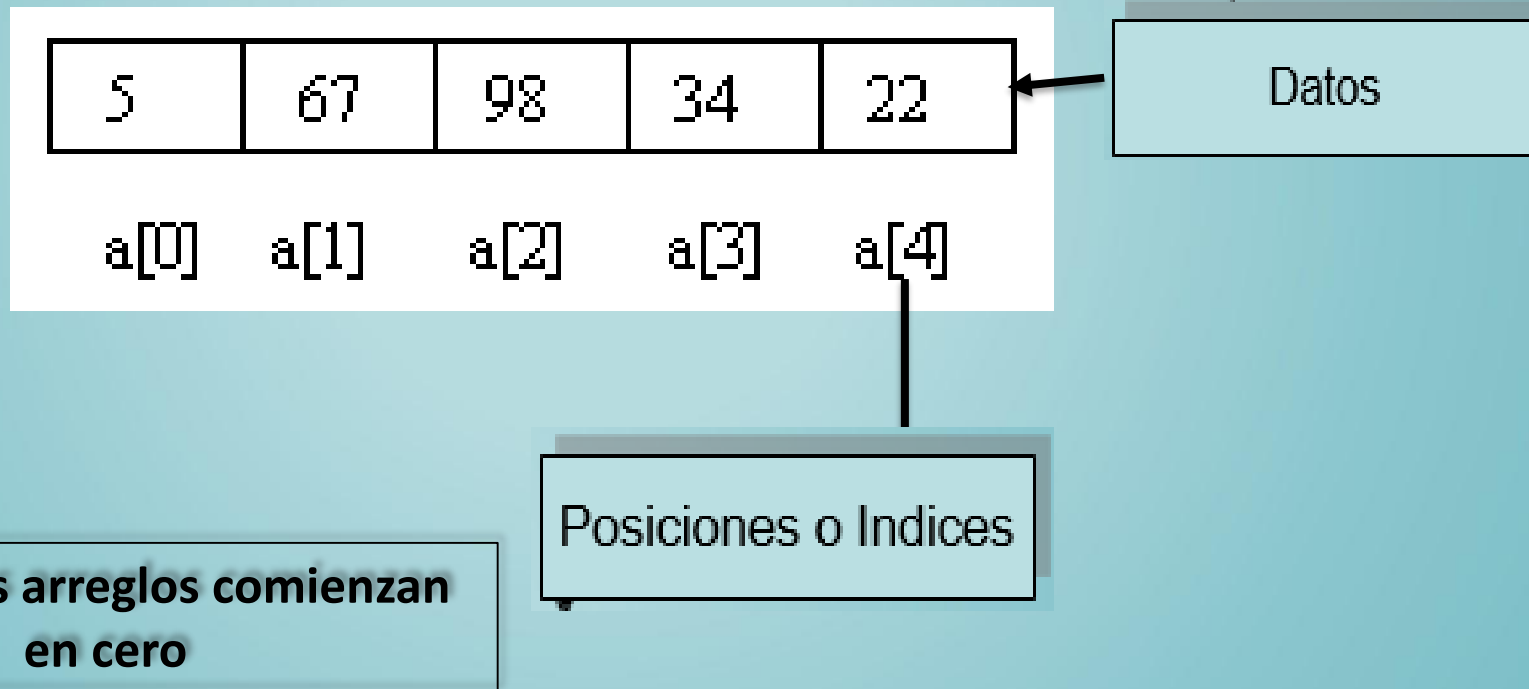


ARREGLOS - ARRAYS



Definición

Los arreglos son una colección de variables del mismo tipo que se referencian utilizando un nombre común.



Definición

Se puede definir de una forma abstracta como “***un conjunto finito ordenado de elementos homogéneos***”.

Por **finito**, entendemos que hay un número específico de elementos en el arreglo; número que debe ser grande o pequeño pero debe existir.

Por **ordenado**, entendemos que los elementos están dispuestos de tal manera que hay un elemento cero, un elemento primero, un segundo, un tercero y así sucesivamente.

Por **homogéneo**, entendemos que todos los elementos del arreglo son del mismo tipo.

Características:

- También se conoce con el nombre de **ARRAY** o **VECTOR**.
- Un arreglo consta de posiciones de memoria contigua de igual tamaño consecutivas que tienen el mismo tipo de variable.
- La dirección más baja corresponde al primer elemento y la más alta al último. El límite inferior se fija siempre en 0 y el superior lo fija el programador .
- Un arreglo puede tener una o varias dimensiones .
- Para acceder a un elemento en particular de un arreglo se usa el nombre del arreglo seguido de un índice entre corchetes.
- La cantidad total de espacio que se usa por un arreglo depende de 2 cosas: El número de elementos en el arreglo y El tamaño del arreglo.

Declaración de un Arreglo Unidimensional

El formato para declarar un arreglo unidimensional es:

nombre_arr [tamaño]:entero

Por ejemplo, para definir un arreglo de diez valores enteros, se debe definir este arreglo de la siguiente manera:

elem[10]:entero;

Esta expresión es la declaración del arreglo. Donde entero es el tipo de datos que almacena el arreglo, elem es el nombre del arreglo, y el número encerrado en los corchetes es el número de valores que contiene el arreglo.

Operaciones con Arreglos

- Cargar un arreglo.
- Recorrer un arreglo.
- Buscar un elemento en particular.
- Acceder a un elemento en una posición determinada.
- Acceder a una posición determinada y mostrar su contenido.
- Insertar un nuevo elemento.
- Eliminar un elemento.
- Ordenar un arreglo

Operaciones con Arreglos

- **Cargar un arreglo.**
- **Recorrer un arreglo para mostrar sus elementos.**

Acción Cargar_Arreglo es

Cargar elementos en el arreglo usando estructura repetitiva.

Ambiente


```
lista[10]:entero;  
l,a:enterio;
```

Algoritmo

```
para (i = 0; i < 10; i:=i+1)  
    escribir("Ingresar el elemento");  
    leer(a);  
    lista[i]:=a;  
fin para  
  
para (i = 0; i < 10; i:=i+1)  
    Escribir (lista[i]);  
  
    si (lista[i]%2=0 escribir ("Es par"));  
    fin si  
fin para  
  
fin acción
```



**CARGA DE
ELEMENTOS EN
EL ARREGLO**



**MOSTRAR LOS
ELEMENTOS
DEL ARREGLO
POR PANTALLA**

ACTIVIDAD-1

Se pide:

- Cargar un arreglo de 10 elementos usando una estructura repetitiva.
- Declarar la dimensión del arreglo usando una constante.
- Mostrar los elementos del arreglo por pantalla.
- Calcular la cantidad de números primos.
- Usar un procedimiento para mostrar por pantalla los elementos del arreglo en sentido inverso.
- Ingresar un entero por teclado e indicar la cantidad de veces que aparece en el arreglo.



ACTIVIDAD-2

Se pide:

- Cargar un arreglo de 10 elementos en forma ordenada ascendente sin elementos repetidos .
- En la etapa de carga se debe verificar que la carga de los elementos ingresados sea ordenado.
- Usa la estructura repetitiva WHILE para cargar los elementos.
- Usar la estructura PARA para mostrar los elementos del arreglo.



Operaciones con Arreglos

- Buscar un elemento en particular.
- Acceder a un elemento en una posición determinada y mostrar su contenido.

escribir("Ingresar posición del arreglo");

leer(a);

si (a < 0) entonces escribir (“Posición no válida”);

sino

si (a >=10) entonces escribir (“Posición no válida”)

sino

escribir (lista[i]);

fin si

fin si

si (a < 0) v (a >=10) entonces escribir (“Posición no válida”);

sino Escribir (lista[i]);

fin si

ACTIVIDAD-3

Se pide:

- Cargar un arreglo de 10 elementos usando sentencia MIENTRAS.
- Mostrar los elementos del arreglo por pantalla.
- Ingresar un entero por teclado e indicar si se encuentra en el arreglo y la cantidad de veces que aparece.
- Ingresar una posición del arreglo y mostrar por pantalla el contenido asociado a la misma.

