

Estructuras

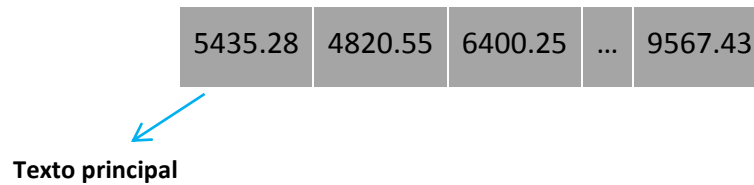
Introducción

Hasta ahora hemos trabajado con vectores, o arreglos, que nos permiten agrupar con un mismo nombre una colección de elementos de un mismo tipo.

Edades

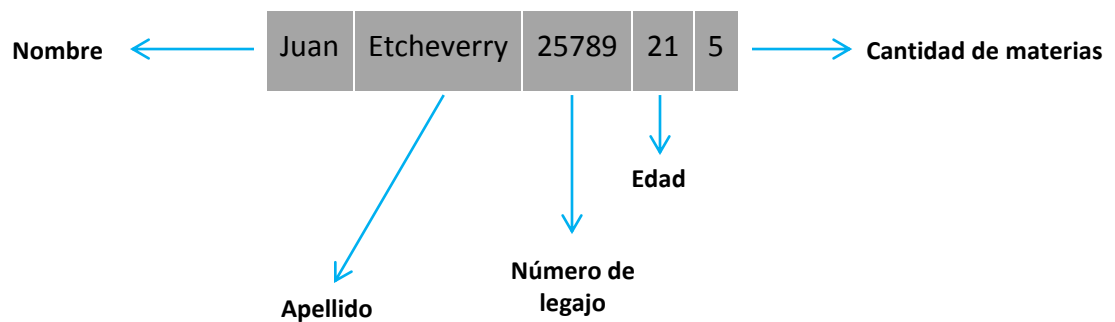


Sueldos



Puede ocurrir que necesitemos trabajar con colecciones de elementos, pero de distinto tipo, como los datos personales de un alumno.

Estructura



Concepto

Una estructura es una colección de una o varias variables, que pueden ser de distinto tipo, agrupadas bajo un único nombre. Esto posibilita su manipulación, por ejemplo, al ser pasadas como argumento o al retornarse de funciones. Se puede usar una estructura para representar un registro de un archivo de empleados: apellido, nombre, documento, legajo, categoría pueden formar una única estructura. Otro ejemplo es para agrupar los distintos componentes de una fecha: día, mes, año.

La sintaxis para declarar un tipo estructura es:

```
struct identificador
{
    tipo1  miembro1;
    tipo2  miembro2;
    ...
}
```

Ejemplo

```
struct fecha {
    int dia;
    int mes;
    int anio;
};
```

A las variables que componen la estructura se las denomina miembros.

MUY IMPORTANTE: Una declaración struct define un nuevo tipo de datos.

Para definir una variable de tipo estructura se escribe la palabra reservada struct seguida del identificador de la estructura y luego el nombre de la variable:

```
struct identificador nombrevARIABLE;
```

Por ejemplo

struct fecha **mifecha** ← la variable **mifecha** es de tipo estructura fecha

Typedef

La palabra reservada **typedef** permite redefinir un tipo bajo otro nombre. Es decir, crea sinónimos. Por ejemplo, podríamos redefinir **int** como “entero” o podríamos redefinir la estructura fecha:

Typedef struct fecha tFecha; ← tFecha es sinónimo de la estructura fecha.

Inclusive se puede declarar la estructura y establecer el sinónimo al mismo tiempo:

```
typedef struct fecha{  
    int dia;  
    int mes;  
    int año;  
} tFecha;
```

Acceso a los miembros de una estructura

Para acceder a los miembros de una estructura se utiliza el punto.

En nuestro ejemplo:

mifecha.dia ← accedemos al día de la estructura.

Pero existe otra forma de acceso a través de un puntero y utilizando el operador → Si p es un puntero a una estructura, accedemos a sus miembros:

p → miembroEstructura

