

# **Tema 9:**

## **Cadenas de caracteres en C**

### **Indice**

- 1. Introducción**
- 2. Cadenas de caracteres. Librería string.h**
- 3. Arrays de cadenas**

# 1. Introducción

- Necesidad de tratar de forma única una palabra o frase (Ejemplo: nombre, apellidos, dirección).
- Una cadena de caracteres es un **array de caracteres** de cualquier longitud que **termina en un carácter nulo**.
- En C el carácter nulo es el '\0'
- En una cadena de caracteres de longitud TAM, la "frase" mas larga que puede contener es de longitud TAM-1

# Introducción

- Cadena de caracteres para almacenar el nombre
  - Longitud máxima: 10
  - Caracteres del nombre más largo que se puede almacenar: 11

A	N	T	O	N	I	O	\0		
---	---	---	---	---	---	---	----	--	--

F	R	A	N	C	I	S	C	O	\0
---	---	---	---	---	---	---	---	---	----

## 2. Cadenas de caracteres. string.h

- **Declaración de una cadena**

**PSEUDOCODIGO**

cad: CADENA[TAM]

**LENGUAJE**

char cad[TAM];

- **Constantes de cadena**

- Se ponen entre comillas dobles “ “
- Automáticamente el compilador le añade el ‘\0’
- No pueden asignarse directamente a una variable con el operador igual, es necesario usar la función de copia de cadenas de string.h ( strcpy )

```
char cad[8];
```

```
cad= “HOLA”; // ERROR !!!
```

```
strcpy (cad, “HOLA”); // CORRECTO
```

H	O	L	A	\0			
---	---	---	---	----	--	--	--

## E/S de cadenas de caracteres

### Funciones específicas para cadenas

- puts(cad); Escribe la cadena en pantalla seguido de un salto de línea
- gets(cad); Lee una cadena de caracteres hasta un ENTER

### E/S con formato

- printf(“%s”, cad); Igual que puts pero sin salto de línea
- scanf(“%s”, cad); Lee hasta un ENTER o un espacio  
(**OJO:** En este caso no hay que poner & porque el nombre de la cadena es ya un puntero al inicio de la cadena)

## Funciones de string.h

- Ver “funciones\_cadena.pdf”
- **IMPORTANTE:** C no tiene operaciones de comprobación de límites de arrays, por eso es responsabilidad del programador (de las funciones que éste diseñe), proteger los arrays frente a posibles desbordamientos o overflows

## 3. Arrays de cadenas

- Es posible declarar un vector de cadena de caracteres
- En realidad equivale a una matriz de caracteres, donde en cada fila se almacena una cadena

Ejemplo:

```
char arraycad[4][8]
strcpy(arraycad[0], "ANA");
strcpy(arraycad[1], "JUAN");
```

A	N	A	\0				
J	U	A	N	\0			