

Fundamentos de Programación

Strings

Facultad de Ingeniería / Escuela de Informática
Universidad Andrés Bello, Viña del Mar.

- arreglo de caracteres
`char palabra[20] = "hola"`
- `#include<stdio.h>`
- `printf(" %s","hola");`

- `strcpy(palabra,"hola")` // se copia *hola* en variable (string)
palabra
- `strlen(palabra)` // retorna el largo (como entero) del string
palabra
- `strcmp(p1,p2)` // valor de retorno es 0 si `p1 == p2`

Concatenar string

- `char p1 = "hola";`
- `char p2 = "mundo";`
- `strcat(p1, " "); // p1 = "hola "`
- `strcat(p1,p2); //p1 = "hola mundo"`

Desarrolle un programa en C que permita ingresar el correo electrónico de cierto usuario, y retorne el nombre de la institución considerando que éste se encuentra ubicado justo antes del último punto.

Ejemplo

Ingrese correo electrónico: bryan@tir.unab.cl

La institución se llama unab

Desarrolle un programa en C que permita al usuario ingresar una contraseña hasta que ésta sea considerada segura. Se considerará que la clave es segura, si tiene al menos una letra mayúscula, una minúscula, un número y un caracter especial.

Ejemplo

Ingrese clave: m1gu3l

No es segura

Ingrese clave: M1gu3l

No es segura

Ingrese clave: M!gu3l

Es segura

Desarrolle un programa que pida el nombre (y apellido) de cierta persona, hasta que éste ingrese **F** Por cada una de estas personas, el usuario indicará cuantos perros y cuantos gatos tiene (en cualquier orden). Una vez que se han ingresado los datos de todos los usuarios, debe indicar el nombre de quien tiene más perros y más gatos.

Ejemplo

Ingrese nombre: Jose Verdugo

Mascotas de Jose: 2 perros 3 gatos

Ingrese nombre: Patricio Silva

Mascotas de Patricio: 3 perros 1 gatos

Ingrese nombre: F

Patricio tiene mas perros

Jose tiene mas gatos

- Permite guardar valores de distintos tipos de datos
- Es posible hacer una instancia de la estructura:

```
struct estructura{ int numero; char  
palabra[10]; }
```
- Es posible definir un tipo de dato a partir de una estructura:

```
typedef struct {int numero; char  
palabra[10]}estructura;
```
- **cuál es la diferencia entre estas dos últimas instrucciones?**

La información de todos los empleados de cierta empresa está almacenada en una variable de tipo `struct` llamada *empleado*, y la información con la que se cuenta corresponde al nombre, edad y sueldo.

Desarrolle un programa en C que lea en un arreglo de estructuras los datos de los N trabajadores de la empresa y que imprima los datos del empleados con menor y mayor sueldo.

Ejercicio 5

A partir de la declaración de las siguientes estructuras realice un programa en C que lea el arreglo *deportistas* y muestre por pantalla los datos (nombre, país, deporte) del atleta que ha ganado mayor número de medallas.

```
struct datos{
char nombre[40];
char pais[25];
}
struct atleta{
char deporte[30];
struct datos pers;
int nmedallas;
}
struct atleta deportistas[30];
```