

# Programación II

# Estructura de datos Registro (struct)

- ▶ Un **struct**, es una estructura de datos en la cual es posible agrupar variables de distinto tipo bajo un mismo nombre, y luego acceder a cada elemento de información de manera individual.
- ▶ Cada una de las variables recibe el nombre de campo; es decir un struct contará con 1 o más campos, de igual o diferente tipo de datos.
- ▶ El conjunto de campos recibe el nombre de **registro**. Estos campos podrán ser de cualquiera de los tipos conocidos, incluyendo arrays, e incluso variables struct de otro tipo definido previamente.
- ▶ En C para poder utilizar un struct, es necesario primero notificar al compilador que se utilizará una estructura compuesta de determinados campos.

# Estructura de datos Registro (struct)

- ▶ En C para poder utilizar un struct, es necesario primero notificar al compilador que se utilizará una estructura compuesta de determinados campos.
- ▶ Como el lenguaje no conoce este “tipo” previamente, hay que definir un “molde” que indique cómo se llamará este tipo de datos compuesto, y cuáles serán los campos que lo compongan.
- ▶ Para acceder a cada campo de un struct se usa la siguiente notación:  
`nombre_variable.nombre_de_campo`

# Estructura de datos Registro (struct)

- ▶ Al igual que con el resto de las variables, es posible pasar un struct por valor o por dirección. Está sujeto a las mismas restricciones que el resto de las variables de tipos primitivos:

- ▶ se puede pasar por valor cuando es una variable simple;

En este caso se dentro de la función, para acceder a un campo en particular del struct se utiliza el operador flecha ->

`nombre_vector->nombre_de_campo`

- ▶ se tiene que pasar por dirección cuando se trata de un vector, o una matriz.