

Semana# 8 ESTRUCTURAS

¿Qué es una estructura?

Es una colección de uno o más tipos de elementos, denominados campos, cada uno de los cuales puede ser un tipo de dato diferente.

Recordemos que un arreglo es un grupo de elementos pero del mismo tipo de datos ya sea entero, void, float, pero todos del mismo tipo de dato.

En las estructuras no todos los elementos pueden ser de un mismo tipo.

Supongamos que queremos almacenar los datos de una colección de discos compactos (CD) de música. Estos datos pueden ser:

Título

Artista

Numero de canciones

Precio

Fecha de compra

```
struct coleccion_CD {  
    char titulo[30];  
    char artista[25];  
    int numCanciones;  
    float precio;  
    char fecha_compra[20];  
};  
  
struct <nombreDeLaEstructura> {  
    <tipoDeDato>    <nombreDeCampo>  
    <tipoDeDato>    <nombreDeCampo>  
    <tipoDeDato>    <nombreDeCampo>  
    ...  
    <tipoDeDato>    <nombreDeCampo>  
};
```

Nótese que **struct** es quien nos abre la estructura y si son observadores al cerrar la estructura **};** tenemos un punto y coma.

Cada estructura va a tener variables, para declarar dichas variables tenemos dos opciones.

- 1) Declaramos las variables dentro de la función principal main().

ESTRUCTURA DE DATOS

(IN-3050; 3 Créditos)

Ing. Alberto Espinoza Zamora email: alberto.espinoza@ucem.ac.cr



```
struct coleccion_CD {  
    char titulo[30];  
    char artista[25];  
    int numCanciones;  
    float precio;  
    char fecha_compra[20];  
};  
  
int main(){  
    struct coleccion_CD CD1, CD2, CD3;  
}
```

Al declarar las variables, cada una de ellas contendrán los campos que se han determinado en la estructura.

- 2) Podemos declarar las variables dentro de la misma estructura , antes de la función principal

```
struct coleccion_CD {  
    char titulo[30];  
    char artista[25];  
    int numCanciones;  
    float precio;  
    char fecha_compra[20];  
} CD1, CD2, CD3;
```

```
#include<iostream>  
#include<stdlib.h>  
using namespace std;  
  
struct Persona {  
    char nombre[20];  
    int edad;  
} Persona1 = { "Alberto", 20 },  
  Persona2 = { "Eduardo", 19 };  
  
int main() {  
  
    cout << "Nombre1: " << Persona1.nombre << endl;  
    cout << "Edad1: " << Persona1.edad << endl;  
  
    cout << "Nombre2: " << Persona2.nombre << endl;  
    cout << "Edad2: " << Persona2.edad << endl;  
  
    system("pause");  
    return 0;  
}
```

Que pasaría si no sabemos los datos de la variable o bien tenemos que pedirle al usuario que los digite. Veamos

```
#include<iostream>
#include<stdlib.h>
using namespace std;

struct Persona {
    char nombre[20];
    int edad;
} Persona1 ;

int main() {

    cout << "Digite su Nombre: \n";
    cin.getline(Persona1.nombre, 20, '\n');
    cout << "Digite su Edad: \n";
    cin>> Persona1.edad ;

    cout << "Mostrando los datos: \n";
    cout << " Nombre: " << Persona1.nombre << endl;
    cout << "Edad: " << Persona1.edad << endl;

    system("pause");
    return 0;
}
```

Estructuras ANIDADAS

Es una estructura dentro de otra estructura

```
struct info_direccion{
    char direccion[30];
    char ciudad[20];
    char provincia[20];
};

struct empleado{
    char nombre[20];
    struct info_direccion dir_empleado;
    double salario;
};
```

empleado				
nombre	dir_empleado			salario
	dirección	ciudad	provincia	

```
#include<iostream>
#include<stdlib.h>
using namespace std;

struct info_direccion {
    char direccion[20];
    char ciudad[20];
    char provincia[20];
};

struct empleado {
    char nombre[20];
    struct info_direccion dir_empleado;
    double salario;
}empleados[2];

int main() {
    for (int i = 0;i < 2;i++) {
        while (getchar() != '\n');//vacía el buffer temporal
        cout << "Digite su nombre : \n";
        cin.getline(empleados[i].nombre, 20, '\n');
        cout << "Digite su direccion: \n";
        cin.getline(empleados[i].dir_empleado.direccion, 20, '\n');
        cout << "Digite su provincia: \n";
        cin.getline(empleados[i].dir_empleado.provincia, 20, '\n');
        cout << "Digite su ciudad: \n";
        cin.getline(empleados[i].dir_empleado.ciudad, 20, '\n');
        cout << "Digite su salario: \n";
        cin >> empleados[i].salario;
    }
    cout << "\n";
    //Mostrando los datos
    for (int i = 0;i < 2;i++) {
        cout << "Nombre: " << empleados[i].nombre << endl;
        cout << "Direccion: " << empleados[i].dir_empleado.direccion << endl;
        cout << "Provincia: " << empleados[i].dir_empleado.provincia << endl;
        cout << "Ciudad: " << empleados[i].dir_empleado.ciudad << endl;
        cout << "Salario: " << empleados[i].salario << endl;
    }
    system("pause");
    return 0;
}
```

ESTRUCTURA DE DATOS

(IN-3050; 3 Créditos)

Ing. Alberto Espinoza Zamora email: alberto.espinoza@ucem.ac.cr



```
/* . Hacer una estructura llamada alumno, en la cual se tendrán los siguientes
Campos: Nombre, edad, promedio, pedir datos al usuario para 3 alumnos, comprobar cuál
de los 3 tiene el mejor promedio y posteriormente imprimir los datos del alumno.*/

#include<iostream>
#include<stdlib.h>
using namespace std;

struct Alumno{
    char nombre[20];
    int edad;
    float promedio;
}alumnos[3];

int main(){
    int mayor = 0,pos=0;

    //Pedir los datos para 3 alumnos
    for(int i=0;i<3;i++){
        while (getchar() != '\n');
        cout<<"Nombre: "; cin.getline(alumnos[i].nombre,20,'\n');
        cout<<"Edad: "; cin>>alumnos[i].edad;
        cout<<"Promedio: "; cin>>alumnos[i].promedio;

        if(alumnos[i].promedio > mayor){
            mayor = alumnos[i].promedio; //sacamos el mayor promedio
            pos = i; // guardamos la posición del mayor promedio
        }
        cout<<"\n";
    }

    //Imprimimos los datos del alumno con el mejor promedio
    cout<<"\nAlumnos con el mejor Promedio: \n";
    cout<<"Nombre: "<<alumnos[pos].nombre<<endl;
    cout<<"Edad: "<<alumnos[pos].edad<<endl;
    cout<<"Promedio: "<<alumnos[pos].promedio<<endl;

    system("pause");
    return 0;
}
```

ESTRUCTURA DE DATOS

(IN-3050; 3 Créditos)

Ing. Alberto Espinoza Zamora email: alberto.espinoza@ucem.ac.cr



Cree dos estructuras una llamada Alumno que contendrá los siguientes campos. Nombre, sexo, edad.

La segunda estructura se llamara Promedio y tendrá los siguiente campos nota1,nota2,nota3.

ANIDE LA ESTRUCTURA PROMEDIO a la estructura alumno.

Pida los datos para "n" cantidad de alumnos, calcule el mejor promedio y muestre sus datos