



Strings - ejercicio 8.3 - Consigna:

Ingresar nombre y DNI de los alumnos de un curso. Como máximo el curso puede tener 50 alumnos. La carga finaliza con un alumno de nombre FIN. Luego de cargar los alumnos se pide:

- a. Ingresar nombres de a uno en uno y buscarlos. Si el nombre está en el curso mostrar su DNI y sino informar que no está. Seguir ingresando nombres hasta que se ingrese un nombre igual a NOBUSCARMAS.
- b. Mostrar el listado de alumnos ordenado alfabéticamente de menor a mayor.

Strings - ejercicio 8.3 - Análisis:

Ambos arreglos se deben manejar de forma paralela

Aquí se muestran en el orden de carga

nombres		DNIs	
0	Zulma		30254565
1	Carlos		25987546
2	Luis		31258444
3	María		22564332
4	Laura		21568566
5	Victor		32156452
6	Ana		34569888



Arreglos ordenados alfabeticamente por nombre

nombres		DNIs	
0	Ana		34569888
1	Carlos		25987546
2	Laura		21568566
3	Luis		31258444
4	María		22564332
5	Victor		32156452
6	Zulma		30254565

Cada DNI debe seguir alineado a su nombre asociado
luego de ordenar los arreglos alfabeticamente por
nombre .

Programa principal:

```
void main()  
{  
    int DNIs[50], cantAlumnos;  
    char nombres[50][21];  
    cantAlumnos = cargarAlumnos(nombres, DNIs, 50);  
    if(cantAlumnos > 0)  
    {  
        buscarAlumnos(nombres, DNIs, cantAlumnos);  
        ordenarAlumnos(nombres, DNIs, cantAlumnos);  
        mostrarAlumnos(nombres, DNIs, cantAlumnos);  
    }  
    printf("Bye!!");  
}
```

Funciones:

```
void leerTexto (char[], int);  
void leerYvalidarNombre(char [], int, char[]);  
int leerYvalidarDNI();  
int cargarAlumnos(char[][21], int[], int);  
int buscarNombre(char[][21], char[], int);  
void buscarAlumnos(char[][21], int[], int);  
void ordenarAlumnos(char[][21], int[],int);  
void mostrarAlumnos(char[][21],int[], int );
```



