

Práctica de Ficheros

En esta práctica ampliaremos el trabajo de la práctica de estructuras en varios aspectos. En primer lugar, definiremos la estructura.

```
struct Alumno{  
    string nombre,apellido1,apellido2;  
    string dni;  
    int edad;  
    float nota;  
};
```

El programa deberá implementar el menú

1. Añadir alumno
2. Imprimir datos alumno (busqueda por dni)
3. Imprimir datos todos alumnos
4. Modificar alumno (busqueda por dni)
5. Eliminar a un alumno (busqueda por dni)
6. Salir

En este caso, deberemos comprobar que el dni sea correcto (concordancia letra/número). Además, el nuevo programa tendrá la siguiente funcionalidad.

Fichero

Nuestro programa ahora deberá guardar la base de datos a un fichero, y leerla desde el fichero al inicio del programa.

La primera vez que se ejecute el programa, no habrá datos para leer. Pero a la salida, se guardará el vector de alumnos. La segunda y posteriores veces que se use el programa, ya se podrá leer el fichero que se guardó la primera vez.

Para que nuestro programa sea realmente útil, queremos pueda ser usado para llevar las base de datos de alumnos de varias clases. Así, que mantendremos un fichero para cada una de las clases que tenemos. Para decidir cuál es la base de datos con la cual queremos trabajar, indicaremos al programa como argumento desde la [línea de comandos](#) la ruta donde se encuentra el fichero a utilizar.

La funcionalidad de leer y escribir a fichero deberá ser implementada en las funciones:

```
1. bool guarda(string ruta_fichero,Alumnos v[],int nalumnos);
```

retorno: Deberá devolver true si se guardo correctamente y false en caso contrario.

ruta_fichero: ruta/nombre donde se guardará el fichero. Ejemplo "alumnos.txt" o "/home/user/alumnos.txt" o "c:\user\alumnos.txt"
Alumnos v[]: vector de alumnos
nalumnos: número de elementos válidos en el vector v

2. `int lee(string ruta_fichero, Alumnos v[], int tam);`

Esta función trata de abrir el fichero. Si es posible abrirlo, lo abre y lee todos sus alumnos y los copia en el vector v. La función devuelve el número de alumnos leídos. tam es el tamaño del vector

Desde el main, se podrá hacer algo así como

```
bool guarda(string ruta_fichero, Alumnos v[], int nal){
    ofstream file;
    ....
}

int lee(string ruta_fichero, Alumnos v[], int tam){
    ifstream file;
    ....
}

int main(int argc, char **argv){
    Alumno v[100];
    int nalum=0;
    if (argc!=2){
        return -1; //no se indico el nombre del fichero
    }
    nalum=lee(argv[1], v, 100);
    int opcion=0;
    do{

    }while(opcion!=6);

    //guardar a fichero el vector
    guarda(argv[1], v, nalum);
}
```