



# Introducción a la Programación

Grado en Ingeniería Informática

## 12. Lectura de datos de archivos

Bloque 4. Recorrido y búsqueda

12. Lectura/Escritura de archivos

Dr. Isidro Verdú

1

### Apertura y cierre de un archivo

#### Apertura

Devuelve el enlace al archivo,  
o **NULL** si se produce un error.

**FILE** fopen (nombre, modo)

El nombre del archivo a abrir

#### Modos de apertura

**r** : lectura

**w** : escritura (sobrescribe)

#### Cierre

fclose (**FILE**)

La variable asociada al archivo

2

## Apertura y cierre de un archivo

El nombre del archivo a abrir (se podría introducir por teclado, claro)

`f` es una variable del tipo `FILE`

Abre archivo modo lectura ASCII

Si se produce un error...

El programa acaba irregularmente

Se cierra el archivo

```
#include <stdio.h>
#include <stdlib.h>
#define NARCH "datos.txt"
main()
{
    FILE *f;

    f = fopen ( NARCH, "r" );

    if (f==NULL)
    {
        printf("Error al abrir archivo %s\n",NARCH);
        printf("Pulsa <enter> ..."); system("Pause");
        exit(1);
    }

    ...
    fclose(f);
    ...
}
```

3

## Lectura de datos en archivos ASCII

### `fscanf`

#### Se usa cuando:

- Se trabaja con archivos ASCII
- Conocemos la estructura de su contenido

datos.txt

```
2.3 15.1 7.0 2.22 13.7
3 5 7
```

4

## Lectura de datos en archivos ASCII

Devuelve el número de datos leídos, o "-1" si se produce un error. Es opcional recoger o no este valor.

La cadena de formato habitual, seguida de las variables a leer, con el símbolo &.

```
int fscanf(FILE, formato, ...)
```

La variable del archivo

```
fscanf(f, "%d %d %d\n", &a, &b, &c);
```

Lee una línea formada por tres enteros separados por espacios. Debe acabar con <intro>

5

## Lectura de datos en archivos ASCII

```
#include <stdio.h>
#include <stdlib.h>
#define NA "datos.txt"
main()
{
    int a,b,c;
    FILE *f;

    f=fopen(NA,"r");
    if (f==NULL )
        {printf("Err"); getchar(); exit(1);}

    fscanf(f,"%d %d %d\n",&a,&b,&c);

    printf("Datos: %d %d %d\n", a,b,c);
    fclose(f);
}
```

6

## Lectura de número cualquiera de datos

Para leer un número cualquiera de datos necesito una función que me detecte el fin de archivo

Devuelve un 0 (false) si no es fin de archivo,  
y un !=0 si es fin de archivo

La variable del archivo

```
int feof(FILE);
```

De este modo puedo leer hasta fin de  
archivo usando la función `feof()`  
¡ojo! No confundir la función `feof` con el  
define `EOF`

```
while (feof(f) == 0)
{
    fscanf(f, "%d", &x);
    ...
}
```

```
while (feof(f) == false)
while ( !feof(f) )
```

7

## Lectura de número cualquiera de datos

Ojo, que con un archivo vacío `feof()`  
debería dar != 0, pero da 0 la  
primera vez.

```
while (feof(f)==0)
{
    fscanf("%d",&x);
    ...
}
```

Se soluciona así:

```
while (!feof(f))
{
    ok=fscanf(f,"%f\n",&x);
    if (ok==1)
    {
        ...sentencias
    }
}
```

8

