

Clase Lunes 10 Marzo de 2008

Ficheros

1.- Declaración de un fichero

Sintaxis:

```
FILE *variable_fichero;
```

ej:

```
FILE *fich;
```

2.- Apertura y cierre de ficheros

- Pasos para utilizar un fichero:

1. Apertura
2. A) almacenar datos
B) obtener datos
3. Cierre

2.1.- Apertura fopen()

- Para abrir un fichero usamos la función fopen()

– Ej:

- `fich=fopen(nombre,modo);`

Fopen() devuelve una dirección que se le asigna a la variable de tipo fichero "fich" que lo indentificará

- Nombre → nombre (y ruta) del fichero
- Modo → tipo de uso que le vamos a dar

Apertura de ficheros

- “r” lectura en un fichero existente (si no existe devuelve un error)
- “w” escritura en un fichero vacío (si existe se destruye)
- “a” escritura al final del fichero (si no existe se crea un nuevo fichero)
- “r+” escritura y lectura (debe existir el archivo)
- “w+” escritura y lectura de un fichero vacío (si existe se destruye)
- “a+” lectura y escritura al final del fichero (si no existe se crea un nuevo fichero)

Ejemplo:

```
f = fopen("C:\\MiFichero.txt", "r");
```

Lectura / Escritura sobre un fichero

- Una vez que tenemos abierto un fichero, lo que procede es leer o escribir de él o sobre él.
- Entre las funciones para leer o escribir datos tenemos:

fscanf-fprintf, fgetc-fputc, fgets-fputs

Conviene usarlas por parejas como aparece ahí

fscanf - fprintf

- Leer y escribir con formato en archivos de texto → fscanf-fprintf
- Los prototipos de las funciones son:

```
int fscanf(FILE *fp, "cadena de control", otros_argumentos);  
int fprintf(FILE *fp, "cadena de control", otros_argumentos);
```

Ejemplo fscanf()-fprintf()

- Ejemplos:

```
fscanf(fichero, "%d", &i);
```

El funcionamiento es similar a scanf ().

En este ejemplo, fscanf lee de fichero un carácter y lo almacena en &i

```
fprintf(fichero, "%d", i);
```

El funcionamiento es similar a printf().

En este ejemplo, fprintf escribe en un fichero el valor de un entero i.

fgetc-fputc

- Leer y escribir caracteres → fgetc-fputc
- Los prototipos de las funciones son:

```
int fgetc(FILE *fp);  
Ejemplo: fgetc(fichero)  
Lee el primer carácter del fichero
```

```
int fputc(int c, FILE *fp);  
Ejemplo: fputc('a', fichero);  
Escribe la letra a en fichero. Notar las ' ', puesto que la función debe recibir un entero, con las ' ' le pasamos el código ASCII
```

fgets-fputs

- Leer y escribir cadenas de caracteres → fgets-fputs
- Los prototipos de las funciones son:

```
char * fgets ( char * str, int num, FILE * stream );  
int fputs ( const char * str, FILE * stream );
```

Ejemplos fgets() y fputs()

- Ejemplo fgets():
`fgets(linea, 81, fichero)`
- Siendo linea un array de cómo mínimo 81 caracteres para evitar desbordamiento de memoria, extraemos los 81 primeros caracteres de fichero
- Ejemplo fputs():
`fputs("Esto es un ejemplo usando \fputs\n", fichero);`
- Insertamos la línea que aparece ahí dentro de fichero.

Cierre de un fichero

- Siempre que acabemos de trabajar con el fichero, debemos de realizar un operación de cierre del mismo. La función que se encarga de ello es `fclose(FILE *fp)`
- *Ejemplo:*
`fclose(fp);`