

Nombre:

Firma:

Equipo:

## Práctica Test 2: Mayo 2004

### 1. Actualizar el archivo de una biblioteca

Una biblioteca tiene una base de datos (archivo binario de acceso directo) donde se recogen el título, los autores y el número de ejemplares de cada libro. Con buen criterio han decidido añadir el código ISBN a su base de datos para facilitar las búsquedas y evitar duplicados.

El bibliotecario se ha dedicado a buscar en Internet los códigos ISBN de todos sus libros y los ha ido guardando en un fichero de texto llamado `isbn.txt`, pero no sabe programar y por lo tanto no puede introducirlos en la base de datos. Nos pide que hagamos un programa que lea `libros.dat` e `isbn.txt`, para crear un nuevo archivo `libros2.dat`.

Afortunadamente la lista de códigos ISBN del archivo `isbn.txt` sigue el mismo orden que los libros almacenados en el archivo `libros.dat`, por lo tanto resulta inmediato asignar cada código al libro correspondiente. Sin embargo, la base de datos original está ordenada al revés (títulos de la Z a la A) y se quiere aprovechar la ocasión para ponerla al derecho (de la A a la Z).

Los archivos de datos están disponible en la dirección `http://www.iit.upco.es/palacios/examen.zip`

### Instrucciones

- 1.1 Firmar la hoja del enunciado y poner el número de equipo en el que estás (por ejemplo PCW07 ó PC07)
- 1.2 Bajar el archivo `examen.zip` con un navegador y extraer `libros.dat`, `isbn.txt`, `examen.c` (opcional) y `mostrar.exe` (opcional) al directorio `c:\temp\examen`

**IMPORTANTE:** Sólo se debe trabajar en el directorio `C:\temp\examen`

El archivo `libros.dat` contiene datos de libros y ha sido creado con la siguiente estructura de datos:

```
#define N 50 /*Longitud de las cadenas */

typedef struct libro {
    char titulo[N];
    char autores[N];
    int num; /* Número de ejemplares en inventario */
    int ndisp; /* Número de ejemplares disponibles */
} LIBRO;
```

- 1.3 Escribir un programa que permita leer registros de `libros.dat` y números de `isbn.txt` y escribirlos en `libros2.dat` que se construirá con la siguiente estructura de datos:

```
typedef struct libro2 {
    char titulo[N];
    char autores[N];
    unsigned long isbn; /* Código ISBN */
    int num; /* Número de ejemplares en inventario */
    int ndisp; /* Número de ejemplares disponibles */
} LIBRO2;
```

- 1.4 Una vez creado el archivo `libros2.dat`, este se debe reordenar de manera ascendente. Teniendo en cuenta que ya está ordenado de manera descendente, esta operación sólo supone un movimiento de registros y no es necesario programar un algoritmo de ordenación. Si se quiere, este apartado puede realizarse en un programa independiente.

### Advertencias

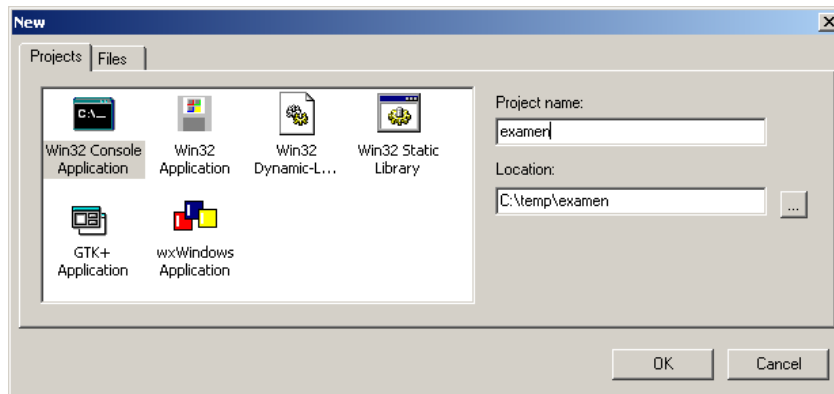
- No se permite utilizar vectores. Todas las operaciones deben realizarse sobre los archivos utilizando las funciones `fscanf`, `fseek`, `fread` y `fwrite`.
- No se puede utilizar `exit`
- Lo que más se valora es que el programa funcione correctamente.
- El programa `mostrar.exe` permite visualizar los contenidos de `libros.dat` y `libros2.dat`

Nombre:

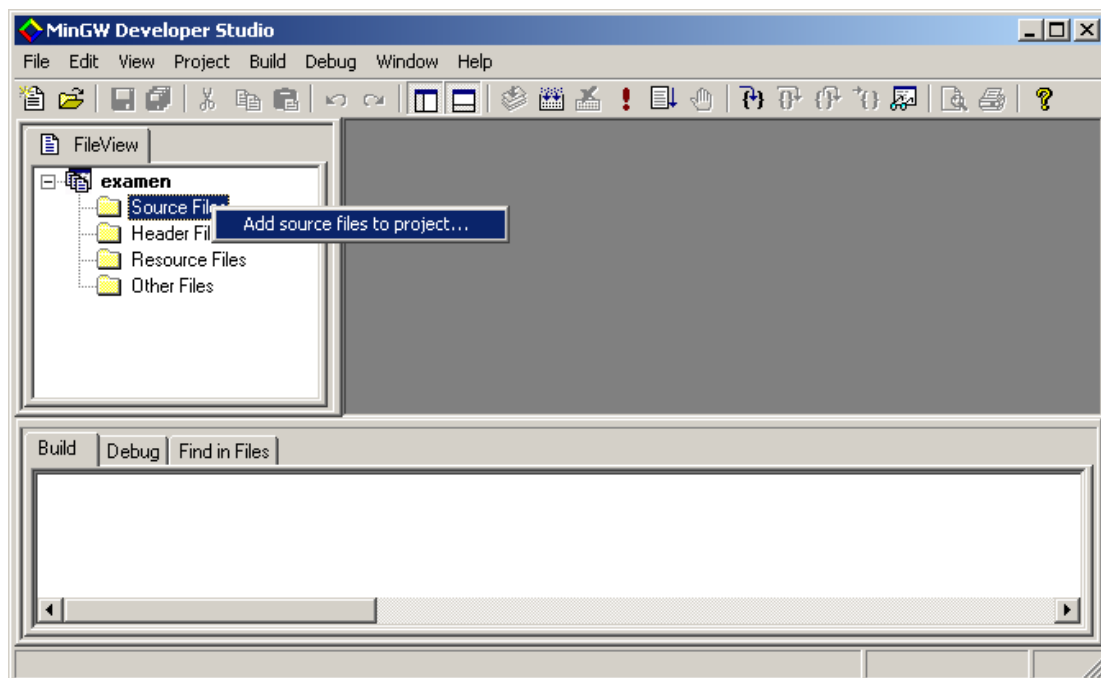
Firma:

Equipo:

Crear el proyecto en c:\temp\examen, con el nombre "examen"



Añadir el archivo examen.c al project



Abrir examen.c para empezar a trabajar (No olvides poner tu nombre)

