

Estructura de datos y de la información

Curso 2003-04

PRÁCTICA 3

Duración estimada: 3 horas

Problema 1

Escribir un pequeño fichero de texto con un editor cualquiera (vi, joe, emacs, ...).

1. Implementar y probar una función que lea todas las palabras del fichero y las escriba por pantalla, cada una en una línea diferente.
2. Implementar y probar una función que lea el fichero anterior carácter a carácter y lo escriba por pantalla del mismo modo.
3. Implementar y probar una función que lea el fichero anterior línea a línea y lo escriba por pantalla del mismo modo.

Problema 2

Escribir un programa que cree un fichero de texto conteniendo los n primeros números pares (n se leerá por teclado). A continuación el programa leerá el contenido del mismo fichero y lo escribirá por pantalla. Visualizar el contenido del fichero con un editor de textos.

Problema 3

Escribir un programa que genere un fichero binario conteniendo los n primeros números pares. A continuación el programa leerá el contenido del mismo fichero y lo escribirá por pantalla.

1. Tras ejecutar el programa, probar a escribir el fichero con la orden Unix *cat*.
2. Probar a abrir el fichero con un editor de texto.
3. Consultar el manual Unix para conocer la utilidad de la orden *od*. Escribir el contenido del fichero con dicha orden. Estudiar el efecto de usar la opción *-a* en la orden anterior: *od -a*
4. Escribir con la orden *od* el contenido del fichero generado en el problema 2.

Problema 4

En este problema vamos a practicar con las funciones básicas de acceso directo a ficheros binarios en C++. Para ello utilizaremos un fichero conteniendo números complejos. Cada número complejo tendrá la siguiente estructura:

```
struct Tcomplejo {  
    double real, imag;  
};
```

1. Escribir una función que lea por teclado números complejos y los almacene en un fichero pasado como parámetro (el fichero, no su nombre).
2. Añadir al programa anterior el código necesario para leer secuencialmente los números complejos almacenados. Escribir los distintos números leídos por pantalla.
3. Añadir una función que se sitúe al final del fichero y, en función de la posición y el tamaño de cada número complejo almacenado, escriba cuántos números complejos contiene el fichero.
4. Añadir al programa anterior el código necesario para que, dada la ubicación relativa de uno de los n números complejos almacenados (1, 2, ..., n), lo lea y lo escriba por pantalla.