

Hands On Lab: Gestió d'entrada/sortida

Juanjo Costa <jcosta@ac.upc.edu>

Sun, 14 May 2023

Contents

Objectius	1
Usar l'interfície d'accés als fitxers	1
Hall of Fame	2
References	3

Objectius

- Entendre i usar l'interfície d'accés als fitxers.
- Entendre i usar el path absolut i el path relatiu.
- Entendre que el contingut dels fitxers són bytes.
- Ser capaç de guardar informació en un fitxer i recuperar-la.
- Ser capaç de buscar i afegir informació en un fitxer.

Usar l'interfície d'accés als fitxers

A classe hem vist l'interfície d'accés als fitxers usant la [SDL], anem a practicar-la.

- Fes un programa (diga-li *write_string.c*) que demani una cadena de caràcters i la guardi en un fitxer "MYSTRING.TXT".

Després d'executar aquest programa...

- Indica el path absolut del fitxer "MYSTRING.TXT" a la teva màquina.
- Indica el path del fitxer "MYSTRING.TXT" relatiu al directori on es troba el teu programa.
- Obre un editor de text (el notepad, per exemple) i obre el fitxer "MYSTRING.TXT". Què hi veus?

- Fes un programa *write_ints.c* que demani 10 números a l'usuari i els guardi en un fitxer "MYINTS.TXT". En particular volem guardar els números en format *binari* ja que ocupa menys que si els guardem en format cadena de caràcters. És a dir, fer alguna cosa de l'estil: `int n; SDL_RWwrite(f, &n, sizeof(n), 1);` Després d'executar aquest programa...
- Obre un editor de text (el notepad, per exemple) i obre el fitxer "MYINTS.TXT". Què hi veus? Oh, sorpresa! El notepad llegeix el fitxer i hi interpreta el contingut com una cadena de caràcters ASCII, però en aquest fitxer no hi ha aquests "caràcters".
- Fes un programa *read_ints.c* que llegeixi el fitxer "MYINTS.TXT" i mostri els enters que hi ha guardats.

Hall of Fame

Un dels problemes que tenen els fitxers és que només permeten afegir informació nova al final, i si intentem "afegir" informació nova al mig, el resultat és que es sobreescriu el contingut antic. Però això és un ús bastant habitual, per exemple, en el nostre context de videojocs, una operació típica és guardar la llista d'usuaris amb puntuacions més altes:

Name	Score

MegaZork	666
DepicableMe	10

De forma que si arriba un nou usuari SuperGirl amb 42 punts caldria que la llista quedés així:

Name	Score

MegaZork	666
SuperGirl	42
DepicableMe	10

- Implementa un programa *hall_of_fame.c* que demani un nom d'usuari (`char*`) i una puntuació (`int`), i ho guardi de forma ordenada (per puntuació) a un fitxer "FAME.TXT". Fixa't que aquí no hi ha un únic tipus de dades sino que es barregen diferents tipus. Els enters tenen mida fixa i per tant sabem exactament què caldrà llegir del fitxer, però les cadenes de caràcters són una seqüència de caràcters de mida variable acabats en un `'\0'`. Si guardem les cadenes literalment, llavors a l'hora de recuperar-les caldrà anar fent lectures del fitxer fins a trobar aquest caràcter. Una opció més eficient seria guardar primer la mida de la cadena mitjançant un enter i després la cadena, d'aquesta forma podríem llegir la cadena del fitxer usant només dues lectures.

References

..[SDL]: <http://wiki.libsdl.org/CategoryIO> “File I/O Abstraction”