

Programma filedistance

Sistemi Operativi Laboratorio Progetto – Settembre 2020

Filedistance è un programma che permette di calcolare la distanza tra 2 file (per distanza si intende il numero di operazioni necessarie per trasformare un file in un altro).

Le operazioni sono le seguenti:

- Aggiungere un byte
- Eliminare un byte
- Modificare un byte

Per calcolare la distanza è stato utilizzato l'algoritmo di *Wagner-Fischer*.

Il programma mette a disposizione 3 funzionalità:

- **Calcolo distanza tra 2 file**

Tramite l'utilizzo del comando "*distance*" con la seguente sintassi:

```
filedistance distance file1 file2
```

si ottiene un output del seguente tipo

```
EDIT DISTANCE: distance  
TIME: tempo di calcolo
```

- **Salvataggio della sequenza di istruzioni**

Tramite l'utilizzo del comando "*distance*" con la seguente sintassi

```
filedistance distance file1 file2 output
```

l'output, sotto forma di istruzioni, viene salvato nel *fileoutput*

- **Applicazione delle istruzioni sul primo file per riottenere il secondo file**

Tramite l'utilizzo del comando "*apply*" con la seguente sintassi

```
filedistance apply inputfile filem outputfile
```

le istruzioni contenute nel *filem* vengono applicate all' *inputfile* e il risultato viene salvato nel *outputfile*

Per sviluppare tali funzionalità sono state utilizzate le seguenti strutture dati:

- La struct "*byte_list_t*" che consente la memorizzazione dei byte contenuti in un file senza conoscerne la lunghezza.

La lista è composta da nodi con al loro interno un blocco di byte (max 256), la size del blocco e un puntatore al nodo successivo.

- La struct *“edit_matrix_t”* che consente la memorizzazione di una matrice di interi che rappresenta la matrice di Edit.
Inoltre, contiene le 2 *“byte_list_t”* contenute i byte dei 2 file da cui è stata generata la matrice.
- La struct *“edit_instruction_t”* che consente la memorizzazione di un’istruzione estrapolata dalla matrice di Edit

Di seguito sono elencate le funzionalità messe a disposizione dalle varie librerie (interfacce pubbliche) che compongono il programma

- ***list.h***

Oltre a dichiarare la struct *byte_list_t* vengono messi a disposizione i seguenti metodi:

- *list_create* che alloca la struct prendendo i byte dal file passato per argomento
- *list_destroy* che dealloca la struct passata per argomento
- *list_get_element* che restituisce il byte corrispondente all'indice passato per argomento
- *list_get_size* che restituisce il numero di byte che compongono il file

- ***edit.h***

Oltre a dichiarare la struct *edit_matrix_t* vengono messi a disposizione i seguenti metodi:

- *matrix_create* che alloca la struct prendendo come parametri i 2 path collegati ai 2 file utilizzando l’algoritmo di Wagner-Fischer
- *matrix_destroy* che dealloca la struct passata per argomento
- *get_edit_distance* che restituisce la distanza di Edit legata ai 2 file a cui appartiene la matrice passata per argomento
- *generate_instructions_file* che memorizza sul file, il cui path è stato passato per argomento, le istruzioni estrapolate dalla matrice di Edit legata alla matrice passata per argomento. L’algoritmo utilizzato risale la matrice dal basso prendendo in considerazione la cella in diagonale, la cella superiore e la cella a sinistra di quella corrente per determinare l’istruzione corretta.
- *apply_instructions_file* che applica le istruzioni contenute
- nel file il cui path è stato passato come secondo argomento al file il cui path è stato passato come primo argomento.
Il risultato viene memorizzato nel file il cui path è stato passato come terzo argomento. L’algoritmo utilizzato ricostruisce il file2 utilizzando come base il contenuto del file1 applicando sopra le modifiche indicate nel *filem_*

- ***edit_instruction.h***

Oltre a dichiarare la struct *edit_instruction_t*, vengono messi a disposizione i seguenti metodi:

- *instruction_create* che alloca la struct prendendo i byte dal file passato per argomento
- *instruction_destroy* che dealloca la struct passata per argomento

Il Build del progetto viene fatto tramite l’uso di uno script *CMake*.