

LZW encodeur

Ce projet permet de compresser et de décompresser des fichiers en utilisant l'algorithme LZW.

Compilation

La commande `make` dans un terminal UNIX permet de générer l'exécutable.

La commande `make clean` supprime tous les fichiers générés par `make`.

Exécution

- `./lzw -e file_in file_out` Comprime le fichier `file_in` dans `file_out`;
- `./lzw -d file_in file_out` Décomprime le fichier `file_in` dans `file_out`.

Etat final

La phase A et la phase B ont été complétées, et la phase C n'a pas été commencée.

L'exécution du code se déroule sur une durée acceptable sur le fichier `big.txt` (6.5MB)

```
// Encoding
$ time ./lzw -e ignore/big.txt ignore/big.comp
Encoding finished successfully.

real    0m1,599s
user    0m1,579s
sys     0m0,020s

// Decoding
$ time ./lzw -d ignore/big.comp ignore/big.out.txt
Decoding finished successfully.

real    0m0,711s
user    0m0,674s
sys     0m0,020s
```

L'algorithme fonctionne aussi sur des fichiers binaires (images, ...).

Il n'y a pas de fuite de mémoire d'après les tests avec `valgrind`.

Répartition du travail

- Alex BAIET : Développement de la phase A;
- Ilyass BERRADI : Développement de la phase B.

Structure

- `lzw.c` : Fonctions d'encodage, de décodage et le main;
- `mapde.h` : Map utiliser pour le décodage;
- `mapen.h` : Map utiliser pour l'encodage;
- `map.h` : Code commun pour les deux structures de map;
- `link.h` : Pour stocker une valeur dans une map;
- `str.h` : Permet de gérer des chaînes de caractères contenant des '\0'.