Instructions pour le projet : Codage linéaire

On vous propose de réaliser un encodeur et un décodeur pour un code correcteur. La théorie de l'algorithme à mettre en oeuvre est expliquée en cours et dans le livre de Vélu. Seulement le premier des chapitres traitant des codes est nécessaire. Il y a aussi de multiples sources d'explications sur internet.

Tâche à réaliser

Il faut écrire un programme qui prend en entré un fichier de données qui contient une suite d'octets encodé avec le code de Hamming H(2,4) augmenté d'un bit de parité. Le fichier encodé est un peu bruité. La matrice génératrice pour ce code est donnée dans le fichier *hamming.c*. Votre programme doit corriger le bruit, décoder le contenu et l'écrire dans un fichier.

Exemple

L'archive tar, accessible ici, contient les fichier

- hamming.c
- myerror.c, myerror.h
- black.c
- transmit.c
- Makefile

L'algorithme du cours (ou du livre) ne spécifie pas complètement l'encodage. Il est formulé est des termes mathématiques. Les détail de l'encodage sont fixés par le code dans *hamming.c*.

Le fichier *hamming.c* contient l'encodeur. Il contient aussi du code que l'on peut utiliser comme exemple pour l'ouverture, la lecture et l'écriture dans un fichier. Par ailleurs, il montre aussi comment on peut gérer la ligne de commande.

Les fichiers *myerror.c*, *myerror.h* contiennent des prototypes et de routines de gestion d'erreurs.

Le fichier *black.c* produit un fichier de données blanc ou noir. On peut regarder

Le fichier *transmit.c* contient le bruiteur. On peut observer le travail du bruiteur en faisant par exemple `./black -s 100 | ./transmit -p 0.1 | hexdump`.

Makefile contient la description pour make. La commande `make` produit les exécutables *hamming*, *black* et *transmit*. Avec l'option `-h` tous ces exécutable montrent leurs paramètres.