

Contrôle continu de l'UE "Systèmes d'exploitation" (HLIN303)

Michel Meynard - Pierre Pompidor

8 novembre 2014

Préambule

Le contrôle continu consiste en un projet découlant directement d'une séance de TD/TP. Il s'agit de créer une application C permettant la compression et la décompression de fichier en utilisant l'algorithme de Huffman (et seulement celui-ci). De plus, l'archivage d'un répertoire pourra être réalisé en Python en utilisant l'application C pour compresser chacun des fichiers du répertoire.

1 Organisation

Les étudiants devront **obligatoirement** travailler en binôme (groupe de 2 appartenant au même groupe de TD) sauf exception justifiée (handicap, étudiant salarié). Le rendu de projet sera effectué à la fin du semestre. Il est demandé lors de cette recette :

- une démonstration de l'application durant 5 minutes (utilisant des fichiers fournis par les examinateurs) ;
- des réponses précises des 2 étudiants aux questions posées durant 5 minutes ;
- un micro rapport papier de 1 à 2 pages décrivant les choix importants effectués ;
- des listings papiers commentés (doxygen) ;

2 Fonctionnalités

Le projet est décomposé en deux programmes C :

huf.c le programme de compression utilisé selon la syntaxe suivante :

```
$huf source dest
```

où **source** est un fichier quelconque et où **dest** est le nom du fichier généré par compression à la Huffman du fichier **source**. Cette compression doit afficher les informations suivantes sur la sortie standard :

- liste des caractères et de leur probabilité d'apparition ;
- arbre de Huffman (tableaux fg fd parent ou par indentation) ;
- affichage des codes de chaque caractère (codeChar(E)=010100) ;
- longueur moyenne de codage ;
- taille originelle de la source, taille compressée et gain en pourcentage comme dans l'exemple suivant :
Taille originelle : 5194; taille compressée : 3761; gain : 27.6% !

dehuf.c le programme de décompression utilisé selon la syntaxe suivante :

```
$dehuf dest
```

où **dest** est un fichier compressé à la Huffman. Le fichier décompressé sera envoyé directement sur la sortie standard.

3 Quelques questions

- Quel est le nombre maximum de caractères (char) différents ?
- Comment représenter l'arbre de Huffman ? Si l'arbre est implémenté avec des tableaux (fg, fd, parent), quels sont les indices des feuilles ? Quelle est la taille maximale de l'arbre (nombre de noeuds) ?
- Comment les caractères présents sont-ils codés dans l'arbre ?
- Le préfixe du fichier compressé doit-il nécessairement contenir l'arbre ou les codes des caractères ou bien les deux (critère d'efficacité) ?
- Quelle est la taille minimale de ce préfixe (expliquer chaque champ et sa longueur) ?
- Si le dernier caractère écrit ne finit pas sur une frontière d'octet, comment le compléter ? Comment ne pas prendre les bits de complétion pour des bits de données ?
- Le décompresseur doit-il reconstituer l'arbre ? Comment ?