

TP n°1

Il s'agit du premier TP de l'année. Il est possible que certaines choses ne fonctionnent pas tout à fait comme indiqué car des ajustements sont effectués sur le système après la rentrée. Ce TP est également très court. Ce ne sera pas le cas des TP suivants.

1) Connexion/déconnexion sur le serveur Linux

Cette partie peut avoir déjà été faite lors du TP de découverte de l'environnement matériel.

a) Démarrage du système Linux local

Le département est équipé de PC fonctionnant avec deux systèmes d'exploitation : Windows 7 et Linux Debian 8. Vous connaissez peut-être Windows, c'est le système le plus répandu pour les ordinateurs à la maison et pour faire de la bureautique. Dans les entreprises de grande taille, le système le plus répandu est Unix. C'est de loin le plus fiable.

Ces deux systèmes sont installés sur l'ordinateur, mais il faut faire un choix au moment de l'allumage. Il faut choisir lequel des deux vous allez utiliser pour la suite. Il n'est pas du tout possible de passer d'un système à l'autre en cours de route.

Nous allons travailler avec Linux. Donc si vous trouvez l'ordinateur allumé avec Windows actif, il va vous falloir redémarrer l'ordinateur. Pour ça, soit vous l'éteignez avec le bouton marche/arrêt en façade (surtout pas en coupant le courant brutalement) ; soit vous cliquez sur un bouton rouge situé en bas à droite de l'écran de connexion et demandez de redémarrer. C'est cette dernière technique qui est la plus recommandée.

Lorsque l'ordinateur démarre, il s'arrête après plusieurs écrans noir et blanc que vous devez laisser défiler, sur une page proposant trois choix : Debian GNU/Linux, Options avancées... et Windows 7. Déplacez la sélection sur le premier et tapez entrée.

Le système Linux démarre, affiche des messages [OK] qui indiquent la mise en route de services et que vous comprendrez entièrement en période 3.

b) connexion au système

A la fin du démarrage, il y a un écran d'accueil. Il faut taper sur entrée jusqu'à voir une zone de texte à remplir avec votre identifiant. Alors il se peut qu'il y ait une anomalie qui sera corrigée prochainement, le système peut demander deux fois votre nom de compte (nom d'utilisateur puis identifiant). Il faut cliquer sur Suivant et non pas appuyer sur entrée. Ensuite seulement il peut demander le mot de passe. Non, ne dites pas que vous avez oublié votre mot de passe ? Ça vous fait une petite visite au CRI chez les administrateurs système de l'IUT.

Lorsque tout est ok, vous êtes connecté(e) au système. Ça ressemble énormément à Windows, mais c'est un autre moteur derrière le capot. C'est un moteur qui n'a aucun virus, cheval de Troie et presque pas de backdoors.

c) Utilisation de l'interface

L'interface que vous avez devant vous est du type fenêtre. Elle ressemble beaucoup à Windows, à part que, par défaut la barre des tâches est en haut, mais ça peut se changer très facilement. Son menu « Raccourcis » permet d'ouvrir des fenêtres sur différents emplacements dans le système : votre dossier personnel, ses sous-dossiers...

Le menu « Applications » permet de lancer des logiciels pour travailler. C'est l'équivalent du menu Démarrer de Windows. Regardez ce qu'il y a. Il est possible de rajouter de nombreux logiciels, mais pas dans les salles de TP.

L'interface de type ligne de commande se démarre en choisissant le menu Applications, puis Utilitaires, puis l'élément Terminal.

d) Déconnexion et/ou extinction

Cela consiste à signaler à l'ordinateur qu'on cesse le travail. Il est indispensable de se déconnecter chaque fois qu'on part, à moins qu'on recherche volontairement les ennuis (piratage du compte).

Il suffit d'aller cliquer sur votre nom de compte en haut à droite et de choisir « Fermer la session » ou « Éteindre ».

Laisser le PC allumé car il est relativement déconseillé de l'éteindre à l'improviste et d'autre part, les postes sont susceptibles d'être utilisés juste après votre départ. Par contre, les PC sont éteints automatiquement la nuit.

2) Connexion au serveur Linux distant à partir de Windows

Il est également possible de travailler avec Linux tout en ayant démarré Windows, mais alors en accédant à un serveur distant. Vous pourrez tester cela à la fin de la séance. Il faudra redémarrer en Windows 7. Ensuite, vous connecter sur votre compte Windows.

Ensuite, dans Windows, le menu Démarrer vous propose une catégorie « Connexion Linux », il suffit de cliquer sur l'item, cela ouvre une session sur le serveur distant ; il s'appelle « calcul ». Vous y retrouverez tous vos fichiers.

Ces fichiers sont également visibles dans la fenêtre Ordinateur, dans le partage réseau appelé M: Le partage Z: contient vos documents Windows. Vous pouvez échanger les fichiers de l'un à l'autre.

3) Éditeur de texte VI (Visual Editor)

a) Principes

i) Introduction

`vi` est l'un des plus anciens éditeurs de texte disponibles avec Unix. Il n'est pas particulièrement ergonomique mais il a le mérite de fonctionner sur la quasi-totalité des terminaux disponibles. Dans votre vie professionnelle, vous serez amené(e) à travailler sur des machines très variées et seul `vi` est vraiment disponible partout.

Une des raisons pour continuer à présenter `vi` (ou `vim`) : il est quasiment le seul à pouvoir être utilisé par les administrateurs système car, pour des raisons de sécurité, pour gérer un ordinateur, on ne peut pas se connecter en mode graphique mais seulement en mode texte (avec `ssh`), et d'autre part, les autres éditeurs de texte enregistrent des fichiers temporaires dans des endroits non sécurisés ce qui peut permettre à des intrus de voir leur contenu. Voir à la fin du sujet pour d'autres éditeurs sécurisés.

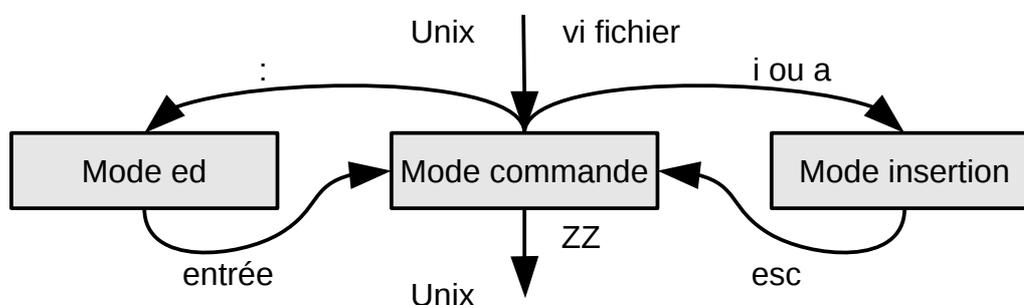
`vi` est un peu déroutant au début. Contrairement aux éditeurs et traitements de textes normaux, on ne peut pas taper du texte directement et se promener à la souris, on doit entrer des « commandes d'édition ».

ii) Modes de fonctionnement

`VI` peut se trouver dans trois modes de fonctionnement :

- le mode commande : dans lequel l'utilisateur entre des ordres (effacer une ligne... voir plus bas), c'est le mode dans lequel on est quand on démarre cet éditeur. Chaque touche (a,i,x...) effectue une action.
- le mode insertion : dans lequel tout ce qu'on tape est inséré dans le texte.
- le mode « ed » : dans lequel on peut entrer des commandes plus complexes telles que chercher un mot, remplacer un mot par un autre. Le passage dans ce mode est provoqué par l'appui sur : (deux-points). En cas de plantage de `vi`, on se retrouve uniquement dans ce mode, il est alors conseillé de quitter puis redémarrer `vi`.

La figure suivante représente les changements de mode de fonctionnement de `vi` :



L'entrée dans `vi` à partir du monde Unix se fait par la commande `vi fichier`. La sortie se fait en tapant `ZZ` en mode commande. Le passage dans le mode insertion se fait en tapant `i` ou `a`, on en sort par

la touche **esc**. Le mode ed démarre par la touche **:** et se termine après l'exécution de la commande d'édition qu'on a mis après le **:**.

ATTENTION : un **A** n'est pas un **a**

b) Commandes d'édition de base

i) Éditer un fichier texte

On démarre l'édition d'un fichier par : **vi fichier**

Mettez un vrai nom à la place du mot *fichier*.

L'éditeur démarre en mode commande, la première page du texte est affichée. Les tildes ~ en début de ligne indiquent que ces lignes sont au delà de la fin du fichier (il n'y a rien).

ii) Quitter l'éditeur

NB : on ne peut quitter l'éditeur qu'en étant dans le mode commande (taper **esc** avant).

On quitte immédiatement l'éditeur sans sauver les modifications par : **:q!**

On quitte l'éditeur en sauvant les modifications dans le fichier par : **ZZ**

On peut enregistrer le texte dans un autre fichier par : **:w nomautre**

c) Commandes d'édition

i) Déplacements dans le texte

Le curseur est géré par les flèches du clavier.

Voici quelques commandes supplémentaires pour se déplacer dans le texte :

:set number affiche les numéros de ligne (**:set nonumber** pour enlever les n^{os}).

:n° va sur la ligne indiquée (utile quand on compile un programme car les compilateurs indiquent les numéros des lignes erronées).

n°G comme la commande précédente, va sur la ligne indiquée par le numéro.

ii) Insérer du texte

i texte... esc insère le texte à droite du curseur, sur la même ligne

a texte... esc déplace le curseur à droite et ajoute du texte, sur la même ligne

o texte... esc insère le texte sur une nouvelle ligne située sous la ligne actuelle

Il existe aussi les commandes **I**, **A** et **O** qui sont intéressantes pour insérer directement en début, fin de ligne ou au dessus. A vous de mémoriser celles qui vous conviennent.

iii) Effacer du texte

x efface le caractère indiqué par le curseur

dd efface la ligne indiquée par le curseur

J regroupe (joint) la ligne courante et la suivante (concaténation de lignes)

iv) Annuler une modification

u annule la dernière opération

d) Commandes élaborées

i) Rechercher un mot

/mot recherche la prochaine occurrence du *mot*

?mot recherche la précédente occurrence du *mot*

n poursuit la dernière recherche

Attention aux caractères **. * [] () ? & ^ \$ **, ils ont un rôle particulier — ils servent à faire des recherches spéciales, on les appelle des *jokers*. Il est prudent de faire précéder chaque caractère douteux par un **** ex : **/3*25\\$ \ (euh\?\) \ .** va chercher les lignes contenant **3*25\$ (euh?)**.

ii) Remplacer un mot (substitution)

:%s/mot/nouveau/g remplace dans tout le texte, *mot* par *nouveau*

:s/mot/nouveau/ remplace dans la ligne courante la première occurrence de *mot* par *nouveau*

:s/mot/nouveau/g remplace dans la ligne courante toutes les occurrences de *mot* par *nouveau*

Le signe % avant s désigne tout le texte, s'il n'est pas mis, ça ne concerne que la ligne courante et le caractère g indique de traiter toutes les occurrences des lignes concernées.

iii) Gestion des blocs

Un bloc est constitué de plusieurs lignes consécutives qui peuvent être effacées ensemble, copiées ou déplacées vers un autre emplacement. Une copie ou un déplacement commence par une mémorisation ou un effacement des lignes en une seule commande. Ensuite, on peut placer le bloc aux différents endroits nécessaires.

nb yy mémorise (copie) un bloc de *nb* lignes commençant sur la ligne actuelle
nb dd extrait et mémorise (coupe) un bloc de *nb* lignes commençant sur la ligne actuelle
p place (colle) un bloc préalablement mémorisé en dessous de la ligne actuelle

4) Exercices avec vi

1°) Tapez le texte proposé en annexe (extrait de "proverbes de programmation" page suivante) ; pour cela, allez sous l'éditeur **vi** en tapant : **vi proverbes.txt** puis tapez ceci :

- o **a** ou bien **i** suivi de la frappe du texte, a ou i vous met en mode insertion à droite ou à gauche du curseur, vous vous habituerez à l'un d'eux.
- o **esc** (échap. sur un clavier français) une fois finie la saisie : vous revenez en mode commande.

2°) Sauvegardez ce que vous venez de taper, par la requête : **w (entrée)**. Votre texte est maintenant sauvegardé dans le fichier *proverbes.txt*, mais vous restez toujours dans l'éditeur.

3°) en ligne 1, on veut remplacer "correctement" par "complètement"

- pour se déplacer en ligne 1, tapez **1G** ou bien : **1 (entrée)** puis avancez sur la première lettre du mot « correctement » en utilisant les flèches du clavier
- pour le remplacer, tapez **R**, frapper le nouveau mot, puis **esc**

4°) ajoutez la phrase " Pour bien programmer il faut : " au début du texte, voici comment faire :

- allez à la première ligne : **1G**
- tapez **O** (o majuscule, pas zéro), regardez ce que ça a fait et veuillez chercher cette touche dans la documentation donnée page précédente
- tapez la phrase puis **esc**

5°) insérez après la deuxième ligne la phrase suivante :

- Utilisez l'étude descendante.

- Pour cela, tapez la touche **o** (la lettre o minuscule pas le chiffre zéro) sur la ligne 2, saisir cette phrase, puis tapez la touche **esc**.

6°) **/Méfiez (entrée)** le curseur se positionne sur le mot Méfiez.

- Supprimez cette ligne "Méfiez-vous ..." avec la commande **dd**, veuillez chercher cette commande dans la documentation donnée plus haut.

7°) Supprimez une autre ligne, n'importe laquelle ou faites un seul autre changement, puis tapez la touche **u** (undo). Si vous faites deux changements, il n'est pas certain que plusieurs usages de **u** vous remette le texte comme avant, ça dépend de la version de vi, donc ne pas abuser et préférer sauver régulièrement.

8°) Sauvez votre texte à ce stade (relire le 2°).

9°) Déplacez les 2 lignes "Soignez ... choisissez ..." ; placez-les après la ligne "Utilisez"

- taper (en aveugle, attention à ne pas taper 22dd ou 222dd, en cas de doute tapez **esc** et recommencez, sinon vous pourrez taper **u**) **2dd** sur la ligne "Soignez ..." (=> 2 lignes coupées)
- taper **p** sur la ligne "Utilisez ..." (=> les 2 lignes collées en dessous)

10°) Copiez les lignes "Ne touchez ... Évitez les astuces" à la fin du texte

- taper (en aveugle) **3yy** sur la ligne " Ne touchez ..." (3 lignes à copier)
- requête **p** sur la ligne "Utilisez ..." (=> 3 lignes collées en dessous)

Veuillez chercher ces commandes dd, yy et p dans la documentation donnée plus haut.

11°) Ajoutez les lignes suivantes à la fin du texte :

Testez la solution à la main avant l'exécution
 Quand le programme est correct, soignez la présentation des résultats

- requête **G** pour aller à la fin du texte
- requête **O** pour insérer les lignes, puis **esc**

12°) sauvegardez votre travail en sortant de l'éditeur (retour sous UNIX):

- requête **ZZ** ou **:wq** (n'oubliez pas de taper esc si vous êtes encore en insertion).

13°) listez le contenu du répertoire (votre dossier personnel) commande **ls**

14°) affichez le contenu du fichier proverbes.txt : commande **more proverbes.txt**

ANNEXE : **proverbes.txt** à taper dans la question 1 de l'exercice précédent.

- Définissez le problème correctement
- Réfléchissez d'abord, vous programmerez plus tard
- Méfiez-vous
- Utilisez des procédures
- Ne touchez pas aux indices de boucles
- Ne recalculez pas une valeur constante dans une boucle
- Évitez les astuces
- Soignez la syntaxe tout de suite
- Choisissez bien vos identificateurs
- Employez des commentaires
- Soignez la présentation
- Posez des questions à l'enseignant : Celui qui pose une question est ignorant cinq minutes, celui qui n'en pose pas est ignorant toute sa vie (Proverbe Chinois)

5) Autres éditeurs

a) Nano

C'est un autre éditeur fiable pour un administrateur. Son mode d'emploi est plus naturel que vi. Il y a une barre d'aide en bas de l'écran.

b) emacs

Cet éditeur est interactif. Il fonctionne comme un traitement de texte classique avec la souris et le clavier. A vous d'explorer ses possibilités (les commandes sont situées dans les menus) pour réaliser les exercices proposés dans la suite. Ce sera l'éditeur que vous utiliserez en algorithmique.

On démarre l'édition d'un fichier par : **emacs nomdufichier**

c) gedit, geany, nedit...

Il existe beaucoup d'autres éditeurs de textes sur Linux (tous ne sont pas installés ici à l'IUT), chacun ses qualités. A vous d'explorer. Ils sont dans le menu qui apparaît quand on clique droit sur un fichier : Ouvrir avec...

Comme les fichiers Linux sont accessibles par M: sur Windows, vous pourriez aussi utiliser notepad pour les éditer, mais ce n'est pas conseillé : parfois, les caractères accentués sont représentés et interprétés différemment selon les machines et parfois la liaison entre Windows et Linux n'est pas fiable, vous pourriez perdre des fichiers. Au moins, employez « Programmer's Notepad » ou « Notepad++ ». Voir la dernière partie de ce TP.

6) Manipulation de fichiers

Dans cette partie, vous allez créer, copier, renommer des fichiers en utilisant les commandes Unix. Vous pourrez voir en parallèle les effets dans une fenêtre, mais l'intérêt de cet exercice est de ne pas utiliser l'interface graphique.

A l'aide de l'éditeur de votre choix (probablement gedit, emacs ou geany), créez un petit fichier appelé **soleil** et contenant les mots « Quel beau temps ! ». Vérifiez à l'aide des commandes **ls** et **more** ou **cat** que ce fichier a bien été créé comme indiqué (son nom et son contenu). Voici les commandes :

```
ls
more soleil
cat soleil
```

Faites une copie du fichier `soleil` avec la commande `cp` appelez cette copie `nuages`. Voici la commande :

```
cp soleil nuages
```

Éditez le fichier `nuages`, modifiez son contenu en « beuh, il va pleuvoir ». Vérifiez l'existence et le contenu des deux fichiers `soleil` et `nuages` à la fois en listant les fichiers (`ls`) et en affichant leur contenu (`more`).

```
more soleil
more nuages
```

Renommez le fichier `nuages`, appelez-le `pluie`. Vérifiez à nouveau l'existence et le contenu des deux fichiers. Vous pouvez aussi vérifier ce qui arrive dans la fenêtre qui montre les icônes.

```
mv nuages pluie
```

```
ls
more pluie
```

Remarquez qu'on peut demander à `more` d'afficher plusieurs fichiers successivement :

```
more soleil pluie
more soleil nuages pluie
```

Voilà donc un message d'erreur de Unix. Regarder avec `ls` pourquoi ce message est apparu.

Maintenant, rajoutez l'extension `.txt` au fichier `soleil`. Pour cela, c'est encore la commande de renommage :

```
mv soleil soleil.txt
```

Vérifiez son résultat avec `ls`.

Faites de même avec le fichier `pluie`. Puis maintenant, changez l'extension du fichier `soleil.txt` en `soleil.lst`. Vérifiez le résultat (dans la fenêtre graphique). Enfin, enlevez cette extension du fichier.

Exercice final : créez un fichier appelé `un` et contenant les mots « est un programmeur » ; créez un second fichier appelé `deux` contenant les mots « qui réfléchit avant de taper. » ; créez un troisième fichier appelé `trois` contenant les mots « un bon programmeur ». Vérifier les noms et contenus des fichiers.

Maintenant le but est d'échanger les contenus des fichiers afin qu'on retrouve le contenu de `un` dans `deux`, celui de `deux` dans `trois` et celui de `trois` dans `un`, sans en avoir ni perdu, ni retapé aucun : uniquement en utilisant des commandes de copie ou de renommage. La commande `cat un deux trois` doit afficher une phrase compréhensible.

```
cat un deux trois
```

```
mv un deux
```

```
cat un deux trois   aïe c'est pas bon, on a fait une bêtise...
```

Rappel : `mv` écrase le fichier destination s'il existe déjà sans prévenir !

7) Encodage des fichiers

Cette partie du TP permet de découvrir des notions sur l'encodage des fichiers, c'est à dire la représentation du contenu sous forme d'octets. On ne pourra pas rentrer dans tous les détails, car vous devinerez que ça peut vite devenir très complexe.

a) Influence de l'extension sur l'application associée

Pour commencer, on va regarder comment l'interface à fenêtre associe un icône à un fichier en fonction de son extension.

1) Ouvrir l'éditeur de texte `geany` (menu Applications, Programmation) et tapez le texte suivant :

```
P6
8 8 127
Mettre n'importe quoi pourvu que ça fasse au moins 8x8x3 caractères en tout.
Blablabla je ne sais pas quoi mettre alors je tape des choses qui me passent
par la tête à ce moment, mais évidemment, ça n'a aucun sens.
```

2) Enregistrez-le dans votre dossier personnel sous le nom `fichier.txt`

3) Avec le menu Raccourcis, ouvrez une fenêtre sur votre dossier personnel afin de voir ce nouveau fichier – si vous ne le voyez pas, c'est qu'il est ailleurs : où l'avez-vous enregistré ? Son icône est celui d'un texte et effectivement c'est bien un texte que vous avez tapé dedans.

4) Quand vous double-cliquez sur ce fichier, il ouvre l'éditeur de texte par défaut, gedit, qui vous montre le contenu précédent. L'extension .txt est associée à gedit par défaut. Si vous voulez ré-ouvrir le fichier avec geany, il faut cliquer droit et choisir ouvrir avec... puis geany.

5) Renommez le fichier en fichier.ppm (il suffit d'appuyer sur F2 quand le fichier est sélectionné, ou de choisir Renommer... dans le menu clic droit sur le fichier). C'est à dire que vous changez de force l'extension (le suffixe). Maintenant l'icône a changé pour une image et si vous double-cliquez, ça va ouvrir un afficheur d'images. Zoomez en avant pour grossir et bien voir l'image. Ce que vous voyez est assez bizarre : le système essaye de comprendre les caractères du fichier comme étant des composantes de couleur, donc ça fait des points colorés de ci de là.

C'est parce que le format PPM n'est pas complexe, il n'est pas compressé et sa structuration est ultra-simple. Le mot P6 signifie que c'est une image de type PPM, 8 8 donnent sa largeur et sa hauteur, 127 indique que les couleurs vont aller de 0 à 127, normalement ce serait 255 mais comme on a mis des caractères, leurs codes sont entre 0 et 100 environ. Ensuite, chaque triplet de codes donne la couleur d'un pixel : rouge, vert, bleu. Ça veut dire que les caractères que vous avez tapés dans le texte sont interprétés en tant que codes d'intensités de couleur. Eh oui, tous ne sont que des nombres dans un fichier : pixels ou texte.

6) Renommez le fichier en fichier.jpg. Mais là, ça ne va plus du tout, le contenu du fichier n'est pas au format JPEG et ne peut pas être affiché. Cependant certains logiciels sont capable de deviner que le nom du fichier ne correspond pas à son contenu et essaient de deviner ce que c'est en regardant les premiers octets.

Ce qu'il faut retenir, c'est la distinction entre le nom du fichier, en particulier son extension qui indique le type du contenu et le contenu lui-même qui est encodé, c'est à dire structuré en suivant une norme précise.

b) Caractères accentués

On va maintenant étudier ce qui concerne le contenu d'un fichier, mais seulement les textes simples. Les textes sont encodés, c'est à dire que chaque signe est représenté par un ou plusieurs octets. Différents codages existent : UTF-8, ISO-8859-15, WINDOWS-1252, etc. Ils viennent de systèmes différents à cause de l'histoire de l'informatique.

1) Pour commencer, relancez geany et tapez les deux lignes suivantes :

Code : ça va être à épeler
en août.

2) Enregistrez-le dans votre dossier personnel sous le nom texte.txt. Ne fermez pas geany, laissez sa fenêtre dans un coin car on va bientôt ré-enregistrer le même fichier avec différents encodages.

3) Ouvrez un terminal (Menu Applications, Utilitaires, Terminal) dans cette fenêtre (menu clic droit dans la fenêtre puis « ouvrir dans un terminal... »). Si vous tapez la commande ls, vous allez voir le fichier texte.txt dans la liste.

4) Tapez la commande more texte.txt. Vous allez voir les lignes que vous avez mises dedans. Tout est ok ? Sinon, vérifiez ce que vous avez fait depuis le début.

5) Tapez la commande od -t aul texte.txt. Elle détaille le fichier : chaque caractère avec son code en dessous. Alors on va regarder de plus près. Quel est le code de la lettre C ? C'est la première du fichier. Quel est le code de o ? Essayez de retrouver ce code plus loin car il y a un autre o. Même question avec le e. Quel est le code du : et du . ? Est-ce que vous avez compris l'idée : chaque caractère = un nombre dans le contenu du fichier. Aïe : quel est le code du ç ? Eh oui, pour certains caractères, il y a deux codes : 195 puis un autre octet.

6) Ce que vous venez de voir est le codage Unicode dans sa variante UTF-8. Nous allons voir un autre codage : le ISO-8859-15. Revenez à la fenêtre geany et choisissez le menu Document puis Définir l'encodage, Européen de l'Ouest, Occidental (ISO-8859-15). Enregistrez le fichier sans changer son nom.

7) Tapez la commande od -t aul texte.txt. Constatez que les codes des caractères accentués et autres (nationaux) ont changé, mais pas les autres. Maintenant, le ç est représenté par un seul octet, 231. Le fichier est donc légèrement plus petit.

8) Par curiosité, tapez la commande more texte.txt. Cette fois, le texte n'apparaît pas bien, il y a des ❖ à la place des caractères nationaux. C'est parce que le terminal est en mode UTF-8 et donc ne peut pas afficher des codes qui sont hors norme. Parfois, au lieu de voir ce caractère bizarre, on voit d'autres lettres.

9) Tapez la commande file texte.txt. Elle va vous dire ce que contient le fichier (par contre elle ne fait pas la différence entre les différents codages ISO).

10) Vous pouvez faire la même manipulation avec le codage WINDOWS-1252. C'est celui qu'emploie Notepad sur Windows. Donc vous saurez quoi faire quand vous avez un tel fichier qui ne s'affiche pas bien sur Linux, ou inversement, pour envoyer un fichier de Linux vers Windows, il faut changer son codage.

c) Codage des fins de lignes

Maintenant, on va regarder ce qu'il en est des fins de lignes car il y a une différence entre Unix, Windows et Macintosh.

1) Dans geany, repassez le fichier en codage Unicode UTF-8.

2) Toujours dans geany, avez-vous remarqué le menu Document, Définir les fins de ligne et ses trois choix. Actuellement, le fichier est en mode LF (Unix). Regardez le résultat de la commande `od -t aul texte.txt` et cherchez comment est représentée la fin des lignes. C'est le code `n\` qui vaut 10 (nl signifie New Line, on l'appelle aussi LF, Line Feed).

3) Maintenant, changez le codage de la fin des lignes par celui de Windows et regardez ce que ça donne comme code. Tiens, il y en a deux. Cr veut dire Carriage Return. La commande `more texte.txt` n'est pas perturbée par ce code supplémentaire.

4) Essayez le codage Mac (mêmes manips). Par contre avec lui, la commande `more texte.txt` ne marche plus du tout, elle ne montre que la dernière ligne. En effet, le code CR dit seulement de revenir au début de la ligne sans en changer.

5) Ce qu'il faudrait faire ensuite, mais il faudrait deux machines pour chacun d'entre vous, c'est de se connecter sur Windows, puis d'aller voir vos documents Linux sur M: et constater que les lignes sont toutes regroupées en une seule et/ou qu'il y a de drôles de signes quand le codage n'est pas bon.

Vous savez maintenant quoi faire pour corriger le codage d'un fichier quand vous l'échangez entre Windows et Linux et que les fins de ligne ne sont pas bien détectées.

d) Ordre des octets et entête Unicode (BOM)

Voici encore une dernière manipulation. Je vous propose de taper un petit texte HTML mais mal encodé et constater comment le navigateur l'affiche (mal). C'est un problème très courant sur internet ou avec les emails.

Lancez geany et tapez le texte suivant :

```
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-15">
  </head>
  <body>
    Code : ça va être à épeler en août.
  </body>
</html>
```

2) Vérifiez que l'encodage est Unicode UTF-8 (et que donc, ça ne correspond pas à ce qui est écrit dans le fichier). Enregistrez-le dans votre dossier personnel sous le nom `texte.htm`. Ne fermez pas geany, laissez sa fenêtre ouverte.

3) Double-cliquez sur le fichier dans la fenêtre où on voit les icônes – ça doit lancer Firefox (navigateur internet). Que voyez-vous ? Je pense que vous avez sûrement déjà vu ce genre de choses en naviguant sur des sites mal construits.

4) Maintenant, rajoutez le BOM Unicode à l'aide du menu Document. Ce sont 3 octets : 239, 187, 191 (comment l'ai-je su ?) qui sont rajoutés au tout début du fichier. Ces codes sont invisibles mais ils indiquent le vrai codage du fichier et du coup, Firefox sait que ce n'est pas de l'ISO-8859. Quand vous faites rafraîchir (CTRL R) dans Firefox, le texte apparaît correctement.

Pour en savoir plus, cherchez Unicode sur Wikipédia.

Nous avons bien conscience que ces manipulations sont trop complexes pour vous, si vous débutez en informatique. Mais dites vous que dans très peu de temps, vous arriverez à les comprendre. Elles ne sont pas à retenir pour le contrôle des connaissances.