

Université de Nice Sophia-Antipolis
Licence de *Mathématiques Appliquées et Sciences Sociales*, deuxième année

Enseignement de *Linux, vecteur de culture informatique*

Pierre Crescenzo (Pierre.Crescenzo@unice.fr),
Jean-Pierre Lips (Jean-Pierre.Lips@unice.fr) et
Isabelle Mirbel (Isabelle.Mirbel@unice.fr)

Travaux Dirigés 10 de la semaine du 30 novembre 2009 : Édition automatique de fichiers textes

1. Introduction

- a. Connectez-vous. Créez votre répertoire de séance `~/Linux-TD/TD10`. Rendez-vous dans ce répertoire.
- b. Récupérez sur le Web les fournitures de ce TD et placez le fichier dans votre répertoire de séance. Désarchivez-le puis effacez l'archive.
- c. Durant tout ce TD, appliquez-vous à utiliser la complétion, l'historique des commandes et les couper/copier/coller au clavier et à la souris.

2. Un psychologue intégré

Lancez `emacs`. Activez le mode `doctor` en tapant `<Méta>-x doctor <Entrée>`. Vous êtes maintenant face à un psychothérapeute informatisé qui va répondre à toutes vos angoisses.

Ce programme est en fait un ancien petit logiciel d'intelligence artificielle nommé `ELIZA`, un des premiers créés, qui a été intégré dans `emacs` par jeu. Il ne faut évidemment pas le prendre au sérieux : ce n'est qu'un programme sans réelle intelligence !

3. Édition automatique de fichiers textes

Utilisez la commande `sed` sur le fichier `f1.don` pour faire les éditions automatiques de texte suivantes.

- a. Remplacez sur les lignes qui contiennent le mot `sed`, la première occurrence de ce mot par `SED`.
- b. Remplacez sur les lignes qui contiennent le mot `sed`, toutes les occurrences de ce mot par `SED`.
- c. Vérifiez avec `grep` que la commande précédente ne porte que sur des mots complets et non des parties de mots (telles `used` ou `suppressed`). Éventuellement, améliorez vos commandes pour qu'elles ne remplacent que les mots complets.
- d. Éliminez toutes les lignes blanches ou vides.
- e. Réduisez tous les espaces consécutifs à un seul espace.
- f. Mettez tous les titres de section du fichier sous la forme `« \section{...} »`. Un titre de section peut se repérer parce qu'il est seul sur une ligne, ne comportent que des lettres majuscules, des espaces, des tirets ou des apostrophes. Mettez les aussi en minuscules.
- g. Remplacez chaque ligne qui commence par `« sed(1) »` par les deux lignes :
SED(1)

- h. Remplacez les cinq espaces du début de certaines lignes par une tabulation (pour taper un caractère de tabulation en `bash`, il faut le faire précéder par un `« Ctrl-v »`).

- i. Les lignes qui commencent par « Sun05 5.5 » contiennent la date de dernière mise à jour et le numéro de page. Modifiez le texte `f1.don` pour que ces lignes ne contiennent que la date de dernière mise à jour cadrée à gauche et la chaîne «Page n» cadrée à droite (pour des lignes de 80 caractères environ).
- j. Dans la section `EXAMPLES` du fichier `f1.don`, et uniquement dans cette section, remplacez les caractères « # » en début de ligne par un seul « % ». (Note : la section `EXAMPLES` est suivie de la section `ENVIRONMENT`.)

4. Script Shell et édition automatique de fichiers textes

L'éditeur `sed` s'utilise aussi fréquemment dans les scripts Shell pour modifier le contenu d'une variable. Dans ce cas, nous utilisons la commande `echo` pour envoyer le contenu de la variable sur la voie standard de sortie qui est traitée par ensuite par `sed` à travers un tube. Le résultat de la transformation peut se récupérer dans le script Shell par une substitution de commandes.

Exemple : Supprimer les trois premiers caractères du contenu de la variable `EXEMPLE` s'obtient par la commande suivante : `EXEMPLE=$(echo $EXEMPLE | sed -e 's/^.//')`

Écrivez un script Shell `ma-substitution` qui change le nom de tous les fichiers transmis en paramètres du script et dont le suffixe est « `c` », en remplaçant tous les caractères souligné « `_` » éventuels par un caractère tiret « `-` ».