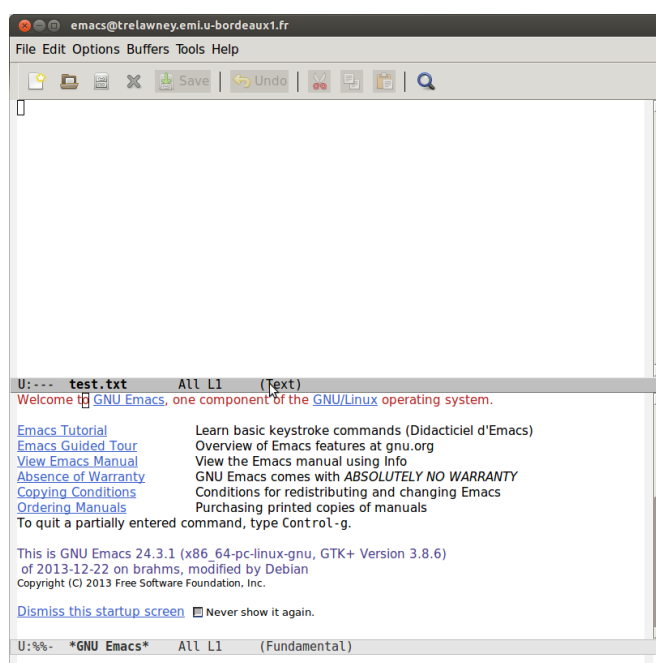


Emacs

1 Principes de base

Créez un répertoire de travail pour ce TP, par exemple `~/UtilisationUnix/emacs/` et placez vous dans ce répertoire. Lancez l'éditeur de texte Emacs en arrière plan pour éditer un nouveau fichier `test.txt` :

```
emacs test.txt &
```



Lors de la première utilisation d'Emacs, la fenêtre est divisée en deux parties. Vous pouvez supprimer la partie inférieure en utilisant le lien `Dismiss this startup screen`.

Remarquez en bas de la fenêtre ce qu'on appelle le mini-buffer, qui sert à emacs pour communiquer avec vous. Lorsque vous tapez une commande, elle s'affiche dans le mini-buffer.

Au dessus se trouve une ligne grisée, la « ligne de mode ». Elle donne des indications telles que le nom du fichier en cours d'édition, le numéro de la ligne où se trouve votre curseur, etc. Comme vous éditez un fichier texte (d'extension `.txt`), le mode est le mode Texte (affiché entre parenthèses).

Pour accéder aux commandes d'Emacs, vous pouvez bien entendu utiliser les menus ou les icônes mais pour une utilisation efficace il est conseillé de maîtriser les raccourcis clavier des commandes principales. La plupart des commandes d'Emacs utilisent les touches `CONTROL` ou `ESC` .

- la touche `CONTROL` s'utilise en combinaison avec un caractère (comme pour la touche `SHIFT`)
- la touche `ESC` doit être relâchée, comme un caractère à part entière.
- dans la suite de la feuille, on utilisera les notations :
 - `C-f` : touche `CONTROL` + caractère `f`
 - `M-f` : touche `ESC` puis caractère `f` ou touche `ALT` + caractère `f` (ce qui se lit *méta-f*).

On appelle **clé** un caractère ou une suite de caractères *attaché* à une commande.

Dans la suite, vous pourrez utiliser la commande **C-g** pour revenir à l'état initial : **C-g** permet **l'abandon du traitement en cours**.

2 Edition de fichiers

Pour le besoin de l'exercice recopiez dans votre répertoire de travail un fichier python du cours *Initiation à l'informatique* du 1er semestre sous le nom `tp01.py`.

Le **Chargement** d'un fichier s'effectue à l'aide de la commande `find-file` liée à la clé **C-x C-f**, ce qui signifie que vous pouvez soit taper **M-x find-file** (appel de la commande), soit taper **C-x C-f** (utilisation de la clé liée à la commande).

2.1 EXERCICE Exemple de chargement de fichier

- tapez **C-x** : Emacs attend la suite ;
- tapez **C-f** : Emacs exécute la commande `find-file` en affichant dans la mini-fenêtre le message


```
Find file: ~/UtilisationUnix/emacs/
```
- possibilité d'obtenir la liste des complétions en tapant **TAB** **TAB** : Emacs ouvre une seconde fenêtre, nommée ***Completions*** pour afficher la liste des complétions ; tapez ensuite le début du nom de fichier suivi de **TAB**
- tapez **RETURN** pour valider la proposition `tp01.py`. La *fenêtre de complétion* est refermée et le chargement du fichier `tp01.py` s'exécute.

Lors du chargement la fenêtre `test.txt` est remplacée par une fenêtre de nom `tp01.py` dans laquelle est affiché le contenu du fichier `tp01.py`. On dit qu'on a ouvert un buffer sur le fichier `tp01.py`. Remarquez que le mode est maintenant (Python) et que les menus disponibles sont différents.

Tapez un caractère : il est inséré dans le buffer mais le fichier n'a pas changé. Lorsque le buffer est différent du fichier, la ligne de mode (au-dessus du mini-buffer) contient deux étoiles :

```
-U:**- tp01.py ...
```

Il faut **sauvegarder régulièrement les modifications**.

La **Sauvegarde** du buffer s'effectue à l'aide de la commande `save-buffer` liée à la clé **C-x C-s**. Lors de la sauvegarde, le mini-buffer affiche un message :

```
Wrote /.../tp01.py
```

2.2 EXERCICE Après la sauvegarde, observez la ligne de mode : les deux étoiles ****** ont disparu. Retapez **C-x C-s**. Quel message s'affiche ?

Tapez **C-x C-w**, puis tapez `tp01-bis.py` dans la mini-fenêtre. Quel est l'effet de cette commande ?

2.3 EXERCICE Revenez au buffer sur `test.txt` (menu **Buffers** ou commande **C-x b**). Ajoutez du texte et sauvegardez.

L'**Insertion** d'un fichier dans le buffer s'effectue à l'aide de la commande `insert-file` liée à la clé **C-x i**.

2.4 EXERCICE Déplacez votre curseur jusqu'à l'endroit où vous souhaitez insérer un autre fichier. Puis tapez **C-x i**. Emacs suit la même procédure que pour le chargement de fichier : il vous demande quel fichier insérer. Testez l'insertion d'un fichier dans `test.txt`.

3 Edition de texte : déplacement et couper/copier/coller

3.1 EXERCICE Déplacements

- essayez **C-*<fleche>*** pour les différentes flèches du clavier

- placez-vous au milieu d’une ligne et essayez les commandes **C-a**, **C-e**, **M-a**, **M-e**.
- **C-l** : recentrage vertical du texte en fonction du curseur et réaffichage
- Déplacez-vous dans le fichier à l’aide de **M->**, **M-<** ou **M-g g** suivi d’un numéro de ligne.

3.2 EXERCICE Suppressions

Insérer, puis supprimer des caractères, des mots, des lignes, en utilisant les clés **C-d**, **M-d**, **M-DEL** et **C-k**.

Copier/coller

- à la souris : on peut sélectionner une zone de texte en maintenant le bouton gauche de la souris appuyé, puis cliquer avec le bouton du milieu à l’endroit où l’on souhaite recopier la zone sélectionnée.
- au clavier : pour sélectionner une région au clavier il faut d’abord déposer une marque au début du texte que l’on veut sélectionner en tapant **C-SPC**, puis il suffit de se déplacer avec les flèches. La région sélectionnée apparait surlignée.
Pour copier cette région tapez **M-w**, pour la couper tapez **C-w**. Le texte coupé est placé dans une sorte de tampon : le *kill ring*. Pour le coller, tapez **C-y**. Il faut noter qu’il y a un historique des "copier" et "couper". Ainsi, **C-y** colle la dernière chose à avoir été copiée ou coupée, vous pouvez taper ensuite plusieurs fois **M-y** et les textes précédents dans l’historique sont collés.

3.3 EXERCICE Effectuez des copier-coller au clavier dans le buffer `test.txt`. Testez l’historique des "copier" et "couper".

4 “undo”

Les clés **C-¹** ou **C-x u** permettent d’annuler l’effet de la dernière commande. Il est possible d’enchaîner plusieurs *undo* et de remonter l’historique des modifications.

4.1 EXERCICE Tester le “undo” sur le fichier `test.txt`. Cherchez comment faire un “redo”.

5 Recherche et remplacement

- la clé **C-s** permet la recherche incrémentale d’une chaîne de caractères que l’on tape dès qu’apparaît dans le mini-buffer l’invite *I-Search* : . Notez que la recherche se fait au fur et à mesure que l’on tape. Chaque **C-s** ensuite va vers la prochaine occurrence de la chaîne dans le buffer (vers l’avant). La touche **Esc** arrête la recherche.
- la clé **C-r** permet une recherche en arrière dans le buffer. On tape la chaîne à rechercher dès qu’apparaît l’invite *I-Search backward* : .
- La clé **ESC %** permet de remplacer différentes occurrences d’une chaîne de caractères par une autre chaîne. Tapez la chaîne à remplacer dès l’invite *Query Replace* : , puis tapez **RETURN**. Tapez ensuite la nouvelle chaîne dès l’invitation *With* : . **Emacs** surligne une occurrence de la chaîne dans le buffer et vous demande si oui ou non il faut la remplacer. Répondre : **y** pour oui ou **n** pour non, pour remplacer toutes les occurrences dans le buffer (sans demande de confirmation).

5.1 EXERCICE Recherchez dans le fichier `tp01.py` les occurrence de la chaîne `def`. Testez également le remplacement des occurrences d’un nom de fonction par un autre nom.

6 Fenêtres

Emacs peut afficher plusieurs «fenêtres», chacune avec un texte différent (une «fenêtre» signifie ici une zone particulière dans la fenêtre d’Emacs).

1. nécessite parfois l’utilisation du “shift” en même temps que du “control”

6.1 EXERCICE Observez le comportement des commandes `C-x 2`, `C-x o`, puis à nouveau `C-x 2`, `C-x o`, `C-x o`, `C-x 3`, `C-x 0`, `C-x 1`. Quel est l'intérêt d'ouvrir plusieurs fenêtres sur le même buffer ? Mettez en pratique en plaçant une fenêtre au début d'un buffer et une seconde à la fin du même buffer.

7 Pour ceux qui sont en avance

7.1 Complétion

La clé `M-/` vous permet de compléter un mot en cours de frappe, si vous l'avez déjà utilisé précédemment dans le buffer. Par exemple si vous devez taper à plusieurs reprises un même nom de fonction ou de variable, la deuxième fois, tapez simplement le début du nom puis tapez `M-/`. Emacs complètera automatiquement. Si plusieurs complétions sont possibles, tapez de nouveau `M-/` et emacs vous fera d'autres propositions.

7.2 Jeux de caractères

Il existe de nombreux jeux de caractères pour encoder ses fichiers.

- 1) Regardez la liste des jeux de caractères disponible dans Emacs en utilisant la commande :
`Option -> Mule (Multilingual Environment) -> Set Coding Systems -> For Saving This Buffer -> SPC`
- 2) De manière générale essayez le plus souvent possible d'utiliser le jeu de caractères `utf-8`². Sélectionnez ce jeu de caractères. Quel est le raccourci clavier permettant de sélectionner un jeu de caractères ?
- 3) Vérifiez dans la ligne d'état d'Emacs que votre fichier est bien encodé en `utf-8`.

7.3 Description des clés, commandes et liaisons

7.1 EXERCICE La commande `describe-bindings` liée à la clé `C-h b` donne la liste des liaisons. Parcourir cette liste, trouvez les différentes commandes d'aide.

7.2 EXERCICE La commande `apropos`, liée à la clé `C-h a`, permet de rechercher une information à partir d'un mot clé. Recherchez une commande qui permet de centrer un paragraphe.

2. Il s'agit des caractères Unicode, cf. www.unicode.org