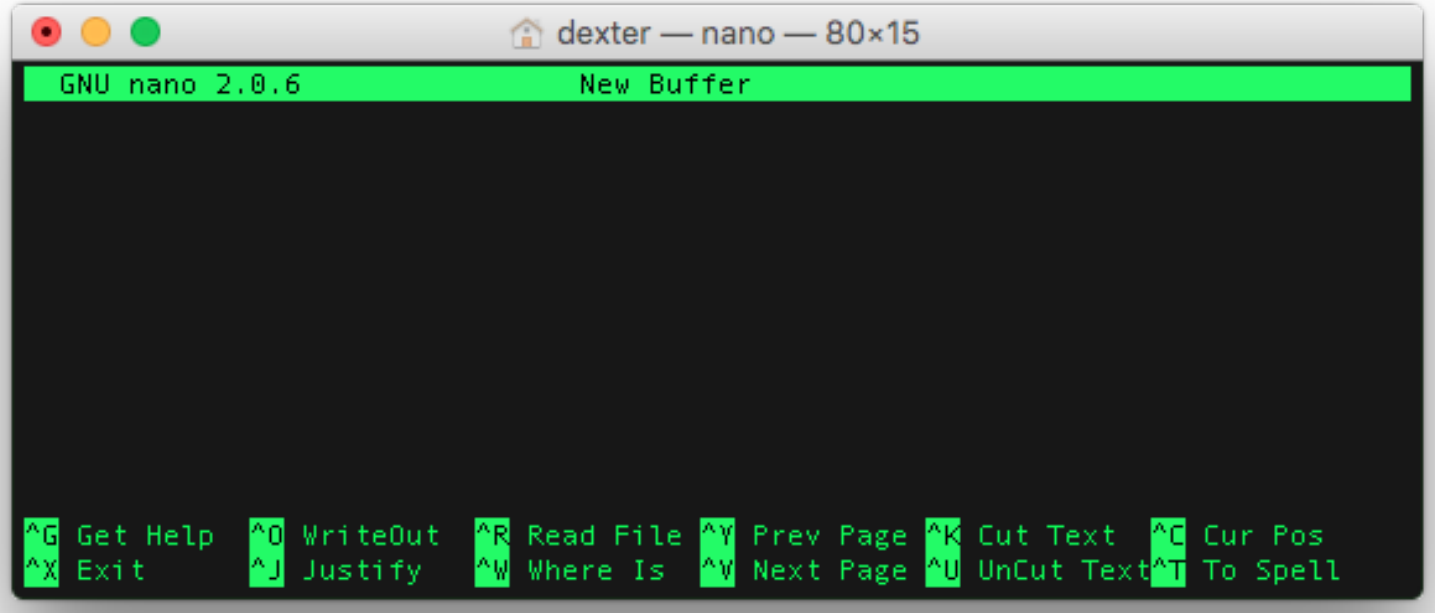
Pratique des machines, installation, utilisation

Utiliser le Shell bash : Initiation aux scripts Shell

Un éditer minimaliste : Nano

$ nano



Commandes possibles indiquées en bas de l'écran.
ˆ représente la touche 'Ctrl'.

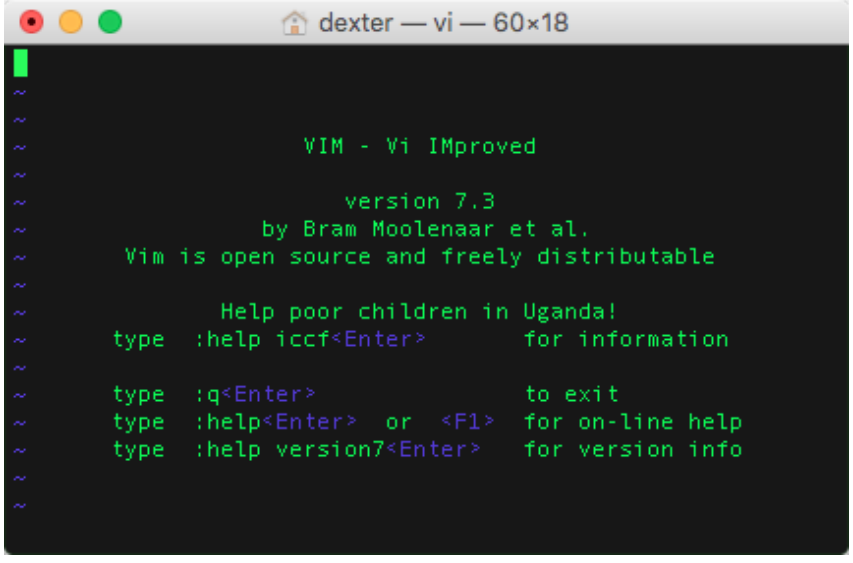
Les éditeurs de code : Vi et Emacs

- Nécessitent un apprentissage
- Conçus pour accélérer ensuite la saisie et l'édition du code
- Entièrement paramétrables
- De véritables IDE
  - saisie avec auto-complétion,
  - mise en valeur des parenthèses correspondantes,
  - coloration syntaxique,
  - compilation dans une sous-fenêtre (comme dans racket),
  - possibilité de programmer ses propres fonctionnalités,
  - ...

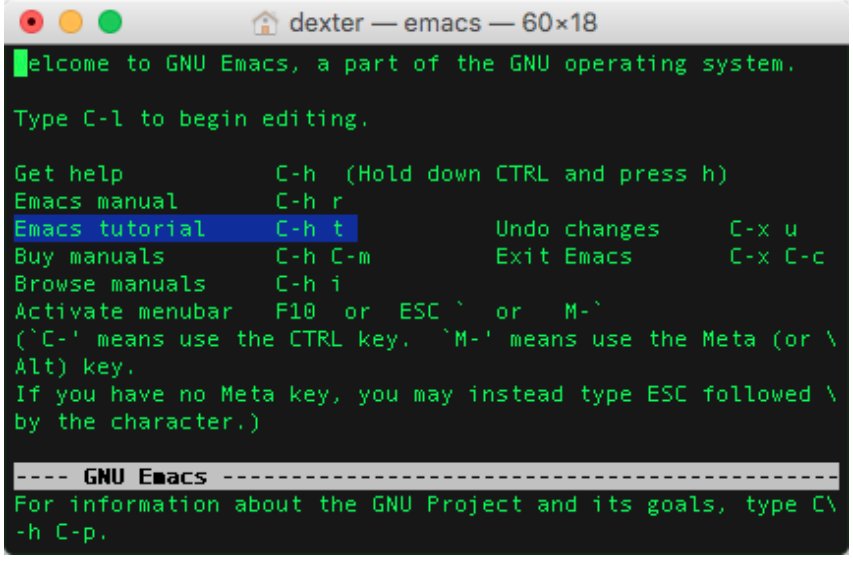
Vi et Emacs : démarrage

Lancer le programme :

$ vi



$ emacs



Lancer le tutoriel :

$ vimtutor

$ emacs puis **Ctrl+h t**

Vi

- Différents ’modes’ de fonctionnement
  - normal : (taper ESC), saisie des commandes
  - ’INSERT’ : insertion de texte
  - ’REPLACE’ : on écrit par dessus le texte existant
- Des commandes composées de 1 ou 2 lettres

Pour gérer les fichiers (taper ESC avant) :

`:help`	avoir de l’aide
`ctrl+w ctrl+w`	passer d’une fenêtre à l’autre
`:q`	fermer une fenêtre
`:q!`	fermer sans enregistrer
`:wq`	enregistrer et fermer
`:w [fichier]`	enregistrer sous le nom [fichier]

Vi : se déplacer

Toujours taper ESC avant pour repasser en mode normal

`h / l`	se déplacer (gauche - droite)
`j / k`	se déplacer (bas - haut)
`e`	la fin du mot courant
`w`	le début du mot suivant
`0`	le début de la ligne
`$`	la fin de la ligne
`gg`	le début du document
`G`	la fin du document

Se déplacer un nombre de fois : `[nombre] déplacement`

`2j`	2 lignes vers le bas
`3w`	3 mots plus loin
`15G`	aller à la ligne 15

Vi : commandes d'édition

Toujours taper ESC avant pour repasser en mode normal

`x`	efface le caractère sous le curseur
`i`	pour insérer des caractères à l'emplacement du curseur
`A`	pour ajouter des caractères en fin de ligne
`R`	remplacer le texte (écrire par dessus)
`v`	enclencher la sélection du texte
`y`	copier
`p`	coller
`d`	efface (delete)
`c`	change (efface et démarre le mode "insertion")
`u`	**annuler la dernière action**
`Ctrl+r`	**remettre ce qu'on a annulé**

Vi : commande et quantificateurs

Associer une commande avec un déplacement :
`opérateur [nombre] déplacement`

`d2$` efface du curseur à la fin de la ligne
`c3w` efface 3 mots et démarre le mode "insertion"

Il existe des raccourcis pour des actions fréquentes :

`dd` efface toute une ligne

Emacs

- Un seul mode de fonctionnement
- Des commandes signalées par les touches spéciales :
  - C => touche Ctrl
  - M => touche "Méta" : Alt ou ESC
  - C-x => appuyer sur la touche Ctrl puis x en même temps

Pour gérer les fichiers :

`C-x C-f`	ouvrir un buffer/fichier
`C-x C-s`	sauver le buffer
`C-x k`	Fermer un buffer
`C-x 1`	une seule fenêtre
`C-x 2`	fenêtre scindée en 2
`C-x C-c`	quitter emacs

Emacs : se déplacer

`C-x C-→`	passer au buffer suivant dans la liste
`C-x C-←`	passer au buffer précédent dans la liste
`C-x o`	passer d'une sous-fenêtre à l'autre
`C-v`	Avance d'un écran
`M-v`	Recule d'un écran
`C-n / C-p`	pour se déplacer (bas-haut)
`C-b / C-f`	pour se déplacer (gauche-droite)
`M-b / M-f`	pour se déplacer d'un mot (gauche-droite)
`C-a`	Va au début de la ligne
`C-e`	Va à la fin de la ligne
`M-<`	Aller à la fin du document
`M->`	Aller au début du document
`M-g g`	aller à la ligne N

Emacs : édition

`C-Space`	commencer une sélection
`Alt-w`	copier
`C-w`	couper
`C-y`	coller
`C-k`	couper toute une ligne
`C-g`	**Annuler la commande en cours**
`C-x u`	**Annuler la dernière modification**

Initiation aux scripts Shell : tests

Test/Condition :

```
if [ ... ]; then
   # faire un truc
elif [ ... ]; then
   # faire un autre truc
fi
```

Sur des nombres	
`$A -lt $B`	`$A` inférieur à `$B`
`$A -le $B`	`$A` inf. ou égal à `$B`
`$A -gt $B`	`$A` supérieur à `$B`
`$A -ge $B`	`$A` sup. ou égal à `$B`
`$A -eq $B`	`$A` égal à `$B`
`$A -ne $B`	`$A` différent de `$B`

`[ $A ] && [ $B ]`	`$A` *et* `$B` *sont vrais tous les deux.*
`[ $A ] || [ $B ]`	*un des deux tests est vrai.*
`[ ! $A ]`	*vrai si* `$A` *est faux.*

Sur des fichiers	
`-e foo`	*le fichier foo existe*
`-d foo`	*foo est un répertoire*
`-f foo`	*foo est un fichier standard*
`-r foo`	*foo est accessible en lecture*
`-w foo`	*foo est accessible en écriture*
`-x foo`	*foo est exécutable*
`foo -nt bar`	*foo est plus récent que bar*
`foo -ot bar`	*foo est plus ancien que bar*

Sur du texte	
`$A = $B`	`$A` *est identique à* `$B`
`$A != $B`	`$A` *est différent de* `$B`
`-n $A`	*la chaine* `$A` *n'est pas vide*
`-z $A`	*la chaine* `$A` *est vide*

Initiation aux scripts Shell : boucles

Boucle 'while' (tant que) :

```
while ... ; do
    # faire un truc
done
```

Exemple:

```
$ x=0
$ while [ $x -lt 10 ]; do echo $x ; x=$(expr $x + 1) ; done
```

Boucle 'for' (pour chaque élément...)

```
for f in ... ; do
    # faire un truc
done
```

Exemple:

```
$ for f in *; do
>    echo $f
> done
$ for i in {1..10}; do echo $i ; done
```

Initiation aux scripts Shell : Saisies utilisateur

Attend une saisie de l'utilisateur et la met dans une variable :

```
read NB
```

Récupérer les arguments sur la ligne de commande

```
$#      # contient le nombre d'arguments
$0      # nom de la commande
$1      # premier argument
$2      # second argument
$3
        # ... etc...
$*      # tous les arguments

$?      # retour de la dernière commande utilisée
```

Les scripts de configuration

`/etc/profile`	réglages pour tous les utilisateurs et pour les shells en général, non spécifiques à bash
`/etc/bash.bashrc`	réglages spécifiques à bash, pour tous les utilisateurs
`~/.profile`	réglages perso, pour les shells en général
`~/.bashrc`	réglages perso, pour le shell bash
`~/.bash_aliases`	importé dans `.bashrc`

Vos scripts de configuration

`.bashrc`

- taille de l'historique
- adaptation de l'affichage à la taille de la fenêtre
- apparence du prompt
- coloration dans le Shell
- coloration du retour de grep ou ls
- définition des alias standards (`egrep`, `l`, `ll`, `la`)
- auto-complétion
- ...

`.bash_aliases`

- tous vos alias, exemples :

```
alias lisp='clisp -q -modern -L french'
alias emacs='emacs -nw'
```

Pratiquons !

Documents à votre disposition sur Moodle

- mémo des commandes élémentaires Vi
- mémo des commandes élémentaires Emacs
- ces slides

- sujet du tp