

**Caractères spéciaux :** . ^ \$ ? \* + [ ] ( ) { } \ ! < > -

>> *.(any char) ^(begin line) \$(endline) \<(begin word) \>(end word)*

→ Echappement : \

→ \* repeter 0 ou plus

→ + repeter 1 ou plus

→ ? repeter 0 ou 1 fois

→ {n} {n,} {n,m} {,m} repeter n \*, au moins n \*, entre n et m \*, 0 ou au plus m \*

→ (p) factorise p

→ p1p2 concat

→ pa | pb OU

→ . (nimporte quel char)

→ [c1c2..cn] (c1|c2| ... |cn) non ordonné

→ [c1-cn] ordonnée

→ [^...] exclusion

→ hors ASCII LC\_TYPE=C (voir man setlocale)

→ [\$^.\*?[(|)-] - perte de sémantique; Except '^' beginning , '-'center

→ [[\]] ⇔ [ ou ]

→ alnum, alpha, digit, lower, upper, blank( \t), space(blank+vertical spaces), ascii, cntrl, graph, print(graph+space), punct, xdigit, word(alnum+\_)

→ [[=e=]] : e é è ê

→ Backreference : ( ... ) \ ( ... \) et \n ; ( ) delimites la portée des opérateurs

→ \b et \< \> inverse de \B et [[:< >:]] → limite de mot.

Exemples: cat, catfish tomcat certificate

\bcat\b → cat ; \bcat → catfish, cat ; cat\b → tomcat, cat

\Bcat\B → certificate ; cat\B → certificate, catfish ;

\Bcat → certificate, tomcat

[[:<:]]cat → cat, catfish

→ \w \W caractère de mot ; \s \S syntaxe c

→ Operateur [[value=~motif]]

→ \${BASH\_REMATCH[0|i]} capture motif complet ou i<sup>eme</sup> sous motif

**Sed OPTIONS 'COMMANDES(Adresse command[args]) FILE.**

**Options:**

- -e CMD ajoute une commande à executer
- -n pas d'affichage des lignes
- -i modif sur place
- -r REGEX
- -f file : lire file qui contient cmds

**Command :**

- Groupement >' cmd;cmd ou {CMD;CMD} ou -e cmd -e cmd
- a\_ppend TEXT / i\_nsert TEXT
- c\_hange TEXT
- d\_elete
- p\_rint
- s\_ubstitute /motif/remplacement/flags;
  - remplacement , backreferences et & <=> motif complet; \U(pper) \L(ower) \E
  - flags : g,p,n,i
- q\_uit <=> break
- n : skip line
- start line ~ number; 0
- y/src/dest <=> tr
- w\_rite file
- = imprime le numero de la ligne lue
- n: remplace le *pattern space* par la ligne suivante; N: \n et ligne suivante to *pattern space*
- h: copier le *pattern space* dans le presse-papier; H : idem mais \n au debut
- g,G : coller
- x: echange pattern <-> space

**Adress :**

number

\$ last

/motif/

Entre deux > adress1,adress2 ; /motif1/,/motif2

Inverse > adress! (equiv grep -v)

adress1~adress2 > Une fois toutes les adress2 en commençant par adress1

adr,+np > Les n lignes après l'adresse

adr,~np > Les lignes après adresses jusqu'à une ligne multiple de n