LETTRES SORBONNE UNIVERSITÉ

Modèles de Linguistique Computationnelle

CM 2 : Bash, un langage utilitaire

M1 Langue et Informatique

Crédits: Gaël Lejeune, Karën Fort, Iana Atanassova, Djame Seddah, Eleni Kogkitsidou, Olga Seminck

---

Alice Millour prenom.nom@sorbonne-universite.fr
Yoann Dupont prenom.nom@sorbonne-universite.fr

2020-2021

Sorbonne-Université

Plan

Expressions régulières

Expressions régulières

expression régulière (*regex*) = *chaîne de caractères, qui décrit, selon une syntaxe précise, un* ***ensemble de chaînes de caractères possibles***

Les regex permettent notamment de décrire/de *matcher* des motifs :

- les adresses e-mail : `[a-zA-Z-]+@[a-zA-Z-]+.[a-zA-Z]{2,6}`
- les mots commençant par une majuscule : `[A-Z][a-z]+` ou `[A-Z]\w+`
- les mots d'au moins 5 lettres : `[a-zA-Z]{6}`
- différentes orthographes d'un même mot : `ex-(a?e|æ|é)quo`
  (« ex-équo », « ex-equo », « ex-aequo » et « ex-æquo »)
- les intitulés de section dans un document : `^Section .+`
  (« Section 1 », « Section 22 », « Section A », etc.)
- etc.

https://fr.wikipedia.org/wiki/Expression_r%C3%A9guli%C3%A8re

Expressions régulières

méta-caractères

caractères qui ne "correspondent pas à eux-mêmes" mais indiquent un traitement à effectuer sur d'autres portions de l'expression régulière.

Par exemple :

le caractère "." représente n'importe quel caractère (dont le point lui-même)

pour faire correspondre le métacaractère à lui-même,

il faut l'échapper = le précéder d'un backslash

échappement de caractères

copiez le texte suivant dans votre éditeur de texte :

" Il n'est vraiment pas en forme... "

et trouvez l'expression correspondant au segment "..."

les méta-caractères (1/3)

- . = tout caractère
- ^ = la séquence qui suit doit être en début de ligne
- $ = la séquence qui précède doit être en fin de ligne

les méta-caractères (2/3)

- [et] = classe de caractères.
  Attention, dans une classe, le caractère ^change de sens, il est utilisé pour la négation de la classe :
  - `[aem]` correspond à `a` `e` ou `m` .
  - `[^aem]` correspond à tout autre caractère
- \= échappe les méta-caractères ou indique une séquence spéciale
- (et) délimitent des groupes de caractères
- | = opérateur logique “ou”
  Attention, priorité faible de l'opérateur :
  - `chaud|froid` correspond à `chaud` ou `froid`
  - `ris|t` correspond à `ris` ou `t`
  - `ri(s|t)` correspond à `ris` ou `rit`

les méta-caractères (3/3)

- * = la séquence qui précède peut apparaître un nombre indéfini de fois
- + = la séquence qui précède doit apparaître au moins une fois
- ? = la séquence qui précède doit apparaître 0 ou 1 fois
  - `vaches?` correspond à `vache` et `vaches`
- { et } = la séquence qui précède doit apparaître entre x et y fois, ex : (abc){2,4} correspond à `abcabc`, `abcabcabc`, et `abcabcabcabc`

Séquences spéciales

- \d = tout caractère numérique ; équivalent à la classe [0-9]
- \D = tout caractère non numérique ; équivalent à la classe [^0-9]
- \s = tout caractère "blanc" ; équivalent à la classe [\t\n\r\f\v]
- \S = tout caractère autre que "blanc" ; équivalent à la classe [^\t\n\r\f\v]
- \w = tout caractère alphanumérique ; équivalent à la classe [a-zA-Z0-9_]
- \W = tout caractère non-alphanumérique ; équivalent à la classe [^a-zA-Z0-9_]
- \b = toute frontière de mot (début de ligne, fin de ligne, ponctuation, caractères “blancs”)

Expressions régulières

Les expressions régulières répondent à une certaine **syntaxe** qu'il faut apprendre...

Tutoriel de prise en main : https://www.lucaswillems.com/fr/articles/25/tutoriel-pour-maitriser-les-expressions-regulieres