

Feuille d'exercices

Partie 2 : Modules, expressions régulières

1 Utilitaires Unix

Exercice 1 Écrire un programme Python reproduisant la commande Unix `cat f1 ... fn`. Cette commande lit n noms de fichiers sur la ligne de commande et affiche leur contenu dans la console.

Exercice 2 Écrire un programme Python `cut.py` reproduisant partiellement la commande `cut`. Votre programme doit être appelé avec la syntaxe `cut.py -d c -f d f`. L'argument c est un unique caractère servant à séparer les champs d'une ligne et d est une description de champs de la forme : $a-b, c-d, e-f, \dots$ indiquant de récupérer les champs d'indice a à b , puis c à d , puis e à f , etc. de la ligne.

Les champs sont numérotés à partir de 1. Par exemple Si un fichier `test.txt` contient :

```
A,B,C,D,E
a,b,c,d,e
```

le résultat de `cut.py -d , -f 1-2,5-5 test.txt` sera

```
A,B,E
a,b,e
```

Dans un premier temps, on supposera que tous les intervalles indiqués dans la commande sont corrects, i.e. que leurs bornes font bien référence à des colonnes du fichier. Dans un deuxième temps, si une borne est trop grande, elle est ramenée à la valeur maximum. Ainsi dans l'exemple ci-dessus 2-10 est équivalent à 2-5.

Les colonnes sont toujours affichées dans l'ordre du fichier et si une colonne est sélectionnée plusieurs fois, elle n'apparaît qu'une fois. Par exemple, le résultat de `cut.py -d , -f 3-5,3-4,1-1 test.txt` sera

```
A,C,D,E
a,c,d,e
```

En effet, 3-5 et 3-4 se recouvrent et sélectionnent 3,4,5. De même, l'intervalle 1-1, bien que donné à la fin, sélectionne la première colonne. L'expression est donc équivalente à 1-1,3-5.

Peut être utile : La fonction prédéfinie `sorted(t)` renvoie une copie triée du tableau `t`. Si les éléments du tableau sont des paires, elles sont comparées par ordre lexicographique.

2 Statistiques

Écrire un programme `stats.py` qui lit un fichier texte et affiche des statistiques sur ce dernier :

- nombre de lignes
- nombre de caractères
- nombre de mots
- mot le plus fréquent (sans être sensible à la casse)
- mot le plus long
- moyenne du nombre de mots par ligne
- lettre la plus fréquente

Peut être utile : On peut tester qu'un caractère (i.e. une chaîne de taille 1) est une lettre avec la méthode `.isalpha()` de la classe `str`.