

Gestion et traitement des données

1. Commandes LINUX

Quelle commande ou quel enchaînement de commandes Linux permet(tent) de réaliser les actions suivantes ?

- afficher seulement les lignes 4 à 6 du fichier `regnes.lst` ;
- afficher seulement les lignes 4 à 6 du fichier `regnes.lst` avec leur numéro de ligne ;
- afficher seulement l'avant-dernière ligne du fichier `regnes.lst`.

Comment afficher en général les lignes numéros n à p d'un fichier ?

2. Script BASH

Rappel : lorsqu'on utilise la commande `ls -aSl -h`, la première ligne affichée est **total xxxxx** où **xxxxx** est la taille totale exprimée dans un format lisible.

Donner le contenu d'un script Bash nommé `grosf.sh` qui

- écrit "Voici les 3 plus gros fichiers" ;
- affiche la liste des informations uniquement pour ces 3 plus gros fichiers ;
- écrit ensuite "pour une taille totale de " ;
- puis qui écrit sur la ligne suivante juste le total – lisible pour un humain – normalement affiché en ligne 1.

Quelle commande permet d'exécuter ce script ?

Quelle commande permet de définir l'alias `grosf` qui exécute ce script ?

3. Script Python

Rapatrier le fichier de données `lea.tsv` et le fichier de script Python3 `regnes.py`. Indiquer ce qu'il faut modifier dans le script pour que l'exécution de ce fichier affiche tout à la fin le texte suivant où la valeur **75** est calculée par le script :

```
Il y a 75 lignes qui ne sont pas des Viridiplantae.
```

Vous commencerez par expliquer ce qu'il faut faire avant de fournir les instructions à rajouter et vous indiquerez l'endroit où les rajouter.

4. Représentations graphiques avec R

On suppose qu'on dispose d'un fichier nommé `valeurs2020.txt` qui contient des données séparées par des espaces, comme pour le fichier `elfgh.txt` vu en cours.

On suppose que les colonne 6 et 7 sont quantitatives, comme pour le fichier `lea.dar` vu en cours.

Donner les instructions **R** qui permettent de :

- lire le fichier `valeurs2020.txt` pour en faire le *data frame* nommé **valeurs** ;
- afficher le résumé statistique des colonnes 6 et 7 seulement de ce *data frame* ;
- trier les données dans le sens croissant sur la colonne 6 ;
- tracer en mode ligne et en rouge la colonne 7 en fonction de la colonne 6 ;
- nommer l'axe des X avec le nom de la colonne 6 et l'axe des Y avec le nom de la colonne 7 ;
- mettre comme titre principal **Tracé des valeurs 2020** ;
- ajouter en bleu clair les points correspondants avec un *plot character* numéro 19.

5. Discussion

Vous essaierez de construire une réponse structurée et bien rédigée à la question suivante, si possible à l'aide d'exemples concrets.

Faut-il connaître très en détail toutes les options de la commande `ls` et de la commande `grep` pour écrire des scripts Bash ?

Il est conseillé d'utiliser au moins trois mots de trois syllabes ou plus pour « transmettre un contenu rédactionnel fort ».

Une dizaine de lignes paraît être une rédaction minimale.

ESQUISSE DE SOLUTION

1. Commandes LINUX

```
# lignes 4 à 6 de regnes.lst

    head -n 6 regnes.lst | tail -n 3

# lignes 4 à 6 de regnes.lst avec numéro de ligne

    grep -n '' regnes.lst | head -n 6 | tail -n 3

# avant-dernière ligne de regnes.lst

    tail -n 2 regnes.lst | head -n 1

# lignes n à p : head -n p | tail -n (p-n+1)
```

2. Script BASH

```
# fichier grosf.sh ; utilisation : sh grosf.sh ou ./grosf.sh (si chmod +x grosf.sh)
# utilisation par grosf si alias "grosf=sh grosf.sh"

echo "Voici les 3 plus gros fichiers"
ls -alS -h | tail -n +2 | head -n 3
echo " pour une taille totale de "
ls -alS -h | head -n 1 | cut -d ' ' -f 2
```

3. Script Python

Il faut utiliser une variable, disons **nbnv** pour compter le nombre de lignes qui ne sont pas des *Viridiplantae*.

Il faut l'initialiser, soit l'instruction `nbnv = 0` sans doute à la ligne 24, avant la lecture du fichier LEA.

Il faut ensuite l'incrémenter dans la boucle de lecture, par exemple après avoir trouvé le règne courant, donc aux environs de la ligne 35, à l'aide des deux lignes

```
if not regneCourant=="Viridiplantae" :
    nbnv += 1
```

Enfin, il suffit d'écrire

```
print("Il y a " + str(nbnv) + " lignes qui ne sont pas des Viridiplantae.")
```

à la fin du script pour obtenir l'affichage final demandé.

4. Représentations graphiques avec R

```
valeurs <- read.table("valeurs2020.txt",head=TRUE,encoding="latin1")
```

```
print(summary(valeurs[, (6:7)]))
```

```
idx <- order(valeurs[,6])  
valeurs <- valeurs[idx, ]
```

```
plot(  
  valeurs[,7]~valeurs[,6],  
  main="Tracé des valeurs 2020",  
  col="red",  
  type="l",  
  xlab=names(valeurs)[6],  
  ylab=names(valeurs)[7]  
) # fin de plot
```

```
points(valeurs[,7]~valeurs[,6],col="lightblue",pch=19)
```

