

Find, grep et les redirections

F. Lassabe

Contents

1	Exercice 1 : find	2
2	Exercice 2: Expressions rationnelles (grep)	3
3	Exercice 3 : combiner find avec une autre commande	3
A	Annexes	5
A.1	Rappel de l'utilisation de find	5
A.1.1	Usage de base	5
A.1.2	Le cas de -exec	5
A.2	Rappel de l'utilisation de grep	5

1 Exercice 1 : find

Pour chaque question qui suit, donner la commande à exécuter.

1. Rechercher les fichiers dont le nom contient la chaîne de caractères toto à partir du répertoire /usr
2. Rechercher les fichiers .h situés sous le répertoire /usr/include.
3. Rechercher tous les fichiers se terminant par .c dans le répertoire /usr
4. Rechercher les fichiers modifiés dans les 3 derniers jours dans toute l'arborescence (/)
5. Rechercher dans toute l'arborescence les fichiers dont la taille dépasse 1Mo
6. Rechercher dans toute l'arborescence, les fichiers appartenant au groupe *dialout*, et dont la permission est fixée à 660
7. Rechercher tous vos fichiers de sauvegarde (terminés par un ~) qui ont plus d'un mois.
8. Rechercher les fichiers et répertoires plus vieux que 3 jours situés sous votre répertoire de connexion.
9. Recherche les fichiers et répertoires dont le nom comprend un s, de taille inférieure à 64 kilobytes, appartenant à root et de type fichier, dont le dernier accès est plus récent que hier
10. Rechercher ensuite les fichiers et répertoires dont le nom commence par une lettre entre p et z dans l'arborescence d'un sous-dossier de votre home, dont la taille est comprise entre 1k et 3k ?
11. Rechercher dans le même sous répertoire les fichiers modifiés il y a plus de 30 jours ?
12. Rechercher dans le même sous répertoire les fichiers modifiés il y a 30 jours ?

13. Rechercher dans le même sous répertoire les fichiers modifiés il y a moins de 30 jours ?

2 Exercice 2: Expressions rationnelles (grep)

Les expressions rationnelles, aussi appelées improprement expressions régulières, permettent de faire correspondre des motifs complexes à des chaînes de caractères. Pour les questions suivantes, complétez la zone de texte avec la commande nécessaire pour y répondre. La majorité des question peut être répondue par l'utilisation d'une commande "pipée" dans grep.

1. Lister tous les fichiers de votre \$HOME qui ont été créés en 2020.
2. Lister tous les fichiers ordinaires de votre \$HOME dont vous possédez les droits de lecture et d'écriture.
3. Lister tous les fichiers dont l'extension est .c
4. Lister tous les fichiers qui commencent par une majuscule.
5. Lister tous les répertoires contenus dans l'emplacement courant.
6. Lister tous les fichiers et répertoires dont le nom commence par un 'p' majuscule ou minuscule dans toute l'arborescence de votre home ?
7. Recherchez ensuite les fichiers et répertoires dont le nom commence par une lettre entre p et z dans l'arborescence d'un sous-dossier de votre home ?
8. Limitez maintenant votre recherche aux objets de type répertoire ?

3 Exercice 3 : combiner find avec une autre commande

Pour les questions ci-dessous, copier la commande nécessaire pour y répondre. Cet exercice se base sur l'usage de l'option *-exec* de la commande *find*.

1. Dans /usr/include, trouver le fichier d'entête qui contient la déclaration de la valeur de Pi, appelée M_PI.

2. Dans `/etc`, lister tous les fichiers se terminant par `.conf`, sans entrer dans les sous répertoires. N'afficher que les noms des fichiers, pas le chemin complet (utiliser `basename`)
3. Dans `/usr/include`, chercher tous les fichiers nommés *types.h*. N'afficher que les noms des dossiers, pas le nom complet (utiliser `dirname`)
4. Dans le répertoire Photos (joint à ce document, sur Moodle), chercher toutes les images du format JPEG, et les convertir en PNG (utiliser la commande `convert`)

A Annexes

A.1 Rappel de l'utilisation de find

A.1.1 Usage de base

Find est une commande de recherche très puissante qui permet de rechercher selon de nombreux critères et d'être combinée avec d'autres commandes. Find s'utilise de la manière suivante :

```
find CHEMIN OPTIONS
```

CHEMIN définit le chemin où find commence sa recherche (par défaut, find descend récursivement dans tous les sous répertoires du chemin de recherche). OPTIONS permet de spécifier les critères de recherche. Parmi les options possibles, en voici quelques unes souvent utilisées :

- name** cette option prend en paramètre un nom ou un motif (à l'image de ls). Elle spécifie le modèles des noms des fichiers et dossiers que find doit trouver.
- type** cette option prend en paramètre un type de fichier à rechercher. Ce peut être par exemple f pour les fichiers simples, d pour les répertoires, l pour les liens, etc.
- perm** cette option prend en paramètre les permissions que doivent avoir les fichiers retournés par find.
- group** cette option prend en paramètre le nom du groupe propriétaire des fichiers que find devra retourner.

Vous pouvez consulter le manuel de find pour plus d'options et d'information sur l'utilisation de cette commande.

A.1.2 Le cas de -exec

Une des options de find est **-exec**. Celle ci permet d'appeler une autre commande sur chaque résultat de la recherche de find. Exec est la dernière option spécifiée lors d'un appel à la commande find. La commande appelée par exec exploitera le résultat en cours grâce à la combinaison de caractères {} et se clôturera par la suite \; Par exemple, rechercher les fichiers dans /var se terminant par .log et afficher leurs informations complètes peut se faire de la manière suivante :

```
find /var -name "*.log" -exec ls -l {} \;
```

A.2 Rappel de l'utilisation de grep

Grep est une commande de recherche de motifs qui retourne les lignes correspondant au motif passé en paramètre. Grep s'exécute de la manière suivante :

```
grep OPTION MOTIF SOURCE
```

où :

- OPTION est facultatif et permet de modifier le comportement de la commande grep
- MOTIF est le motif (sous forme d'expression rationnelle) à rechercher dans la source
- SOURCE est le fichier contenant le texte dans lequel rechercher les motifs.

Les options intéressantes peuvent être :

- H** pour afficher le nom du fichier dans lequel le motif courant a été trouvé.
- v** pour inverser la sortie de grep (afficher les lignes qui **ne** contiennent **pas** le motif)
- i** pour ignorer la casse dans le motif (pas de distinction maj/min)

Par exemple, la commande **grep -H Bob *.list** va rechercher dans tous les fichiers .list les lignes contenant le mot Bob et les afficher ainsi que le nom du fichier associé.