

Évaluation

Durée : 50 minutes

Répondez directement dans ce fichier en complétant les blancs (signalés par un surlignage jaune).

Envoyez ce document complété en pièce jointe à aline.hufschmitt@iedparis8.net en précisant « REGROUPEMENT IED SEPTEMBRE 2017 » comme sujet du mail.

Nom	Prénom	Numéro étudiant

Culture générale (1 points)

Que signifie l'acronyme LUG (en anglais et en français) ?

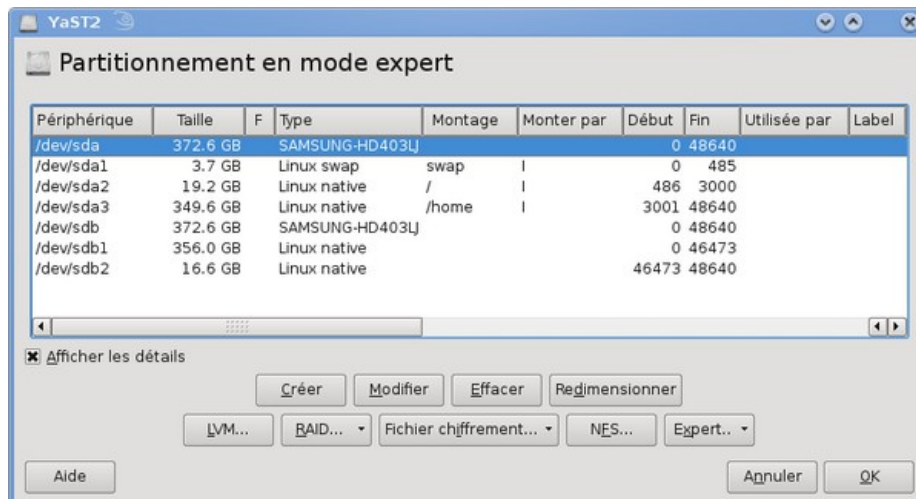
Que signifie le sigle LTS (en anglais et en français) ?

Qu'est-ce que le "lorem ipsum" ?

Quelle est la différence entre un éditeur de texte et un traitement de texte ?

Dans un document Latex, j'ai placé des ancres avec la balise `\label{}` et j'y fait référence avec `\ref{}` plus loin dans le texte. Après compilation de mon document, toutes les références sont remplacées par des "??". Quelle erreur ai-je commise ?

A quoi sert le fichier `.bashrc` (en trois mots) et où s'attend on à le trouver ?



D'après l'image ci-dessus, sur cette machine, combien y a-t-il

- de disques ? _____
- de partitions par disque ? _____
- de quel type sont ces partitions ? _____

Utiliser le manuel (2 points)

Il existe plusieurs fonctions de la librairie permettant de manipuler les expressions régulières (regex). Comment trouver les pages de documentation sur ces fonctions de la librairie (et seulement elles) à partir du mot clef « regex » ? indiquez la commande et le résultat obtenu.

\$ _____

Quelle commande saisir pour trouver des informations sur la commande "locate" dans le manuel ?

D'après la manuel à quoi sert cette commande ?

La commande a besoin de lire des informations dans une base de donnée, d'après le manuel, comment créer cette base ?

Les licences (1 point)

Pour chacune de ces licences, indiquez brièvement pour quel usage elle a été conçue principalement (ce qu'elle protège ou ses particularités si elle en a).

Licence	Que protège t'elle ?	Est-elle copyleft ?
GNU-GFDL		Oui / Non
CeCILL		Oui / Non
BSD		Oui / Non

Que signifie copyleft ?

Quelle est, grossièrement (le sujet est vaste, inutile de trop détailler), la différence entre « libre » et « open source » ?

Processus (1 points)

Complétez les blancs en indiquant les raccourcis claviers ou les commandes appropriées :

Pour stopper proprement l'interpréteur Python ou Lisp :

Pour stopper en force un processus qui tourne en boucle :

Pour mettre un processus en pause à l'arrière plan :

Pour examiner les ressources consommées par les processus : \$

Pour récupérer le PID d'un processus tournant en arrière plan dans la console :
\$

Pour arrêter en force un processus planté (PID=23452) :
\$

Se déplacer, manipuler les fichiers (5 points)

Indiquez les commandes à saisir :

Pour se déplacer dans son répertoire personnel : \$ _____

Pour revenir dans le répertoire parent : \$ _____

Pour créer les fichiers `tic`, `tac` et `toe` : \$ _____

Pour renommer le fichier `tic` en `le_tic` : \$ _____

Pour supprimer le fichier `tac` : \$ _____

Pour créer un dossier nommé `test` : \$ _____

Pour créer un chemin `test/script/shell` : \$ _____

Pour savoir où on est : \$ _____

Pour lister les fichiers du répertoire courant en affichant les fichiers cachés : \$ _____

Pour effacer le répertoire `script/shell` qui contient des fichiers : \$ _____

Pour supprimer le répertoire vide `test` : \$ _____

Pour visualiser le contenu du fichier texte `~/ .bashrc` : \$ _____

Pour connaître le type du fichier `bin/date` : \$ _____

Pour comparer deux fichiers `tic` et `toe` : \$ _____

Si je suis dans le répertoire `/home/damien/documents` et que je veux aller dans `/home/arthur/images`, indiquez la commande à saisir pour m'y rendre

- avec un chemin absolu : _____

- avec un chemin relatif : _____

Liens, redirections, pipes (2 points)

Indiquez les commandes à saisir :

Pour créer un raccourci nommé `doc` vers `/home/arthur/Document` :

\$ _____

Pour afficher le message `bonjour` et le rediriger dans un fichier nommé `message.txt` :

\$ _____

Pour mettre à jour la liste des paquets disponibles dans les dépôts avec `apt` et rediriger la sortie d'erreur vers un fichier `log.txt` :

\$ _____

Pour exécuter le script `script.sh` situé dans le répertoire courant, rediriger la sortie vers un fichier `result.txt` et les erreurs au même endroit :

\$ _____

Pour faire la même chose, mais sans écraser le contenu du fichier :

\$ _____

Pour trier le fichier `listing.txt`, supprimer les lignes en double, et afficher les 10 dernières lignes (en une seule commande) :

\$

Permissions, propriété, chemins d'accès (4 points)

Indiquez les commandes à saisir :

Pour donner la propriété du fichier `secret.sh` à l'administrateur et à son groupe habituel :

\$

Pour voir les droits des fichiers du répertoire courant :

\$

D'après la ligne suivante :

```
drwxr-xr-- 6 arthur staff 204 26 jui 10:11 WWW
```

Quel est le type de fichier ?

Qui a le droit de modifier ce fichier ?

Comment créer un lien physique vers ce fichier ?

Qui n'est pas autorisé à voir le contenu de ce fichier ?

Indiquez la commande à saisir pour interdire à Arthur de modifier ce fichier, sans modifier le reste :

Pour savoir quel Shell utilise Arthur, dans quel fichier puis-je aller voir ?

Si je crée ma propre commande `date` placée dans le répertoire courant, comment lancer cette commande pour être sûr que c'est elle que j'exécute et non celle du système ?

Quelle est la base de l'arborescence d'un système de fichier GNU/Linux ?

A quoi sert la variable `PATH` de l'environnement ?

Expressions régulières (2 point)

Voici un listing contenu dans le fichier `listing.bas` :

```
10 INPUT "Quel est votre nom ? "; NOM$
20 PRINT "Bonjour "; NOM$
30 INPUT "Combien d'étoiles voulez-vous ? "; NOMBRE
40 LET ETOILE$ = ""
50 FOR I = 1 TO NOMBRE
60 LET ETOILE$ = ETOILE$ + "*"
70 NEXT I
80 PRINT ETOILE$
90 INPUT "Voulez-vous plus d'étoiles ? "; ETOILE$
100 IF LEN(ETOILE$) = 0 THEN GOTO 90
110 LET ETOILE$ = LEFT$(ETOILE$, 1)
120 IF (ETOILE$ = "0") OR (ETOILE$ = "o") THEN GOTO 30
130 PRINT "Au revoir ";
140 FOR I = 1 TO 200
150 PRINT NOM$; " ";
160 NEXT I
170 LPRINT
```

Ecrivez une commande avec `grep` pour récupérer les lignes commençant par les nombres de 50 à 130 inclus :

\$

Ecrivez une commande avec `grep` pour récupérer toutes les lignes contenant un point d'interrogation :

\$

Ecrivez une commande avec `grep` pour récupérer toutes les lignes contenant le mot `PRINT` (mais pas `LPRINT`) :

\$

Ecrivez une commande avec `grep` pour récupérer toutes les lignes s'achevant par un nombre :

\$

Script Shell (2 points)

Ecrire une commande Shell qui compte à rebours de 10 à 0 en attendant une seconde (environ) entre chaque nombre affiché (indice : commande `sleep`) :

\$

Écrire un script Shell qui simule un jeu de Quizz avec une seule question. Voici un exemple du comportement attendu pour votre script (en bleu les saisies de l'utilisateur) :

```
$ ./test.sh
L'ASM est un langage assembleur (vrai/faux) ? vrai
bravo !

$ ./test.sh
L'ASM est un langage assembleur (vrai/faux) ? faux
eh non !
```

