Semaine 8 - BASH

Gestion d’objets avec BASH – 20%

# But du laboratoire

* Comprendre et manipuler les fichiers et dossiers en BASH.
* Automatiser l’exécution des scripts avec CRON.
* Gérer les installations de logiciels avec un script interactif.
* Appliquer les conditions et boucles dans des scripts avancés.
* Implémenter une tâche finale combinant plusieurs concepts.

# Date de remise

Dimanche, 16 mars 2025 - 23h55

# Documents à remettre

Un dossier compressé sur Léa contenant ;

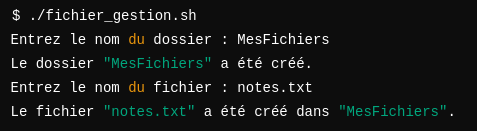
* fichier\_gestion.sh
* install\_logiciel.sh
* supprime\_fichiers.sh
* planifie\_backup.sh
* gestion\_systeme.sh
* Une capture d’écran de la tâche CRON (backup)

### **Question 1 : Manipulation de fichiers et dossiers**

**Énoncé**  
 Écrivez un script fichier\_gestion.sh qui :

1. Demande à l’utilisateur un **nom de dossier** et crée ce dossier s’il n’existe pas.
2. Demande un **nom de fichier**, puis crée ce fichier dans le dossier sélectionné.
3. Vérifie si ce fichier existe, et si oui, affiche sa taille et la date de dernière modification.

**Exemple d’exécution :**

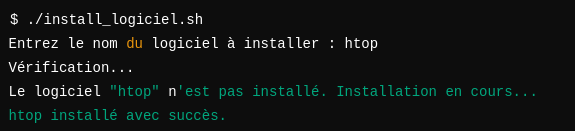


### **Question 2 : Installation automatisée de logiciels**

**Énoncé**  
 Créez un script install\_logiciel.sh qui :

1. Demande à l’utilisateur un logiciel à installer.
2. Vérifie si le logiciel est déjà installé.
   1. Si non, installe le logiciel en fonction du gestionnaire de paquets disponible (apt, pacman, dnf).
3. Affiche un message de confirmation de l’installation.

**Exemple d’exécution :**

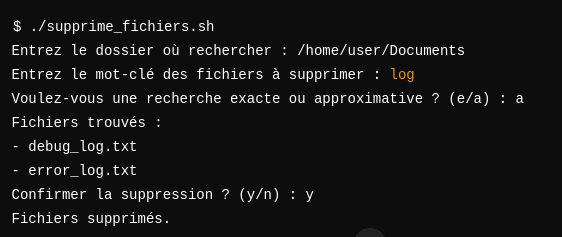


### **Question 3 : Suppression conditionnelle de fichiers**

**Énoncé**  
 Écrivez un script supprime\_fichiers.sh qui :

1. Demande à l’utilisateur un dossier où rechercher.
2. Demande un mot-clé pour la suppression.
3. Demande si l’utilisateur souhaite une suppression **exacte** ou **approximative** (\*mot\*).
4. Affiche les fichiers trouvés avant de les supprimer, avec confirmation.

**Exemple d’exécution :**

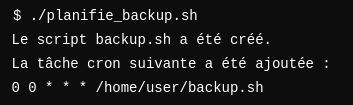


### **Question 4 : Automatisation avec CRON**

**Énoncé**  
 Créez un script planifie\_backup.sh qui :

1. Vérifie si un fichier /home/user/backup.sh existe.
2. Si non, crée un script backup.sh qui archive /home/user/Documents. Par archive nous parlons d’une copie du dossier. Cette dernière n’a pas besoin d’être compressé. La copie peut être faite dans n’importe quel autre dossier que documents.
3. Ajoute une tâche CRON pour exécuter ce script **tous les jours à minuit**.

**Exemple d’exécution :**

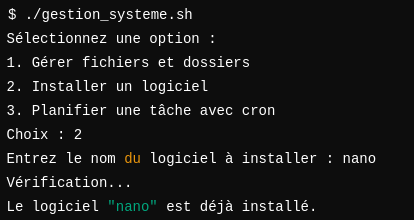


### **Question 5 (Finale) : Intégration des concepts**

**Énoncé**  
 Écrivez un script avancé gestion\_systeme.sh qui combine plusieurs concepts vus :

1. Propose à l’utilisateur 3 options :
   1. (1) Gérer des fichiers et dossiers.
   2. (2) Installer un logiciel.
   3. (3) Planifier une tâche avec CRON
2. Selon le choix de l’utilisateur, exécute le script correspondant (fichier\_gestion.sh, install\_logiciel.sh, planifie\_backup.sh).
3. Si l’utilisateur choisit une option invalide, affiche un message d’erreur.

**Exemple d’exécution :**



### **Grille de correction**

|  |  |  |
| --- | --- | --- |
| Critère | Description | Points possibles |
| Fonctionnement | Le script s'exécute correctement et sans erreurs. | /3 |
| Respect des exigences | Le script répond aux consignes de l'examen (ex : arguments, fichiers, etc.). | /3 |
| Logique de fonctionnement | Les étapes du script sont bien structurées et logiques (ex : utilisation efficace des boucles et conditions). | /3 |
| Optimisation du code | Le script évite les redondances et utilise des solutions efficaces. | /3 |
| Organisation et lisibilité | Le code est clair, bien structuré et espacé correctement. | /2 |
| Commentaires | Présence de commentaires expliquant les parties importantes du script. | /1 |