

Mini Projet

Système de favoris pour le shell

On se propose de réaliser un script *bash* permettant de créer un petit système de « favoris » pour vos répertoires importants.

1 Préliminaires

Date de rendu : mardi 28 mars 2023 à 23h59

1.1 Généralités

- Le rendu du projet se fera *UNIQUEMENT* sur le site <https://lms.univ-cotedazur.fr/2022/course/view.php?id=7308§ion=6> sous la forme d'une archive (.zip, .tar, .rar, .tgzetc.) qui contiendra l'intégralité du code ainsi qu'un fichier INSTALL/README contenant les indications pour compiler et exécuter votre programme ... Votre projet devra contenir :
 - Le code source de votre projet, ainsi que tout autre fichier dont sa bonne exécution en dépend (e.g. fichier PNG, etc.) ;
 - Un fichier INSTALL/README, qui indiquera clairement comment compiler et exécuter votre programme.
- Si possible, vous allez également devoir rendre un rapport d'environ une page minimum et deux pages maximum.
 - Vous y expliquerez ce que votre projet fait, ainsi que la liste des options implémentées.
 - Vous justifierez les choix que vous aurez effectués durant l'élaboration de votre programme (structures de données, algorithmes ...) et discuterez les limites de votre implémentation.

1.2 Remarques

La note du projet prendra en compte :

- La propreté et la lisibilité du code ainsi que tout ce qui facilitera sa compréhension par les correcteurs (noms des variables, commentaires, modularité, etc).
- La facilité d'utilisation du code (le fichier INSTALL/README est-il lisible et suffit-il pour utiliser le code ?). La compilation de votre code devrait marcher sur tout type de machine - pensez à tester avant la rendue finale.
- La correction et la performance du code.
- La qualité de la rédaction du rapport.

Mini Projet

1.3 Instructions

Le projet doit répondre à toutes les instructions obligatoires.

- Il est strictement interdit de copier du code du projet de vos collègues. Attention : l'obfuscation du code n'est pas une solution, c'est très facile à détecter !
 - Plagiat interne et externe : il n'est bien sûr pas interdit de dialoguer entre vous sur ce que vous faites, ni de demander à Google ou à ChatGPT ce qu'il en pense, mais la copie est interdite et vous devez citer vos sources. Tout plagiat détecté sera passible d'un envoi en commission disciplinaire de l'UNS.

1.4 Ce que fait un bon programmeur : rappels

Même si cela peut paraître évident, voici quelques règles indispensables à la survie d'un programmeur.

- Toute fonction écrite doit être testée !!!!!
- Un test contient généralement la ligne de commande avec ses arguments pour lancer le test AINSI QUE le résultat qui s'affiche à l'écran. Si c'est un affichage graphique, une impression d'écran conviendra.
- Le code doit être organisé en différents fichiers au contenu cohérent.

2 Cahier des charges

2.1 Préambule

Lorsque nous travaillons dans un terminal, certains répertoires reviennent très souvent. C'est par exemple le cas de notre dossier personnel. Le shell offre un raccourci pour se déplacer dans le dossier personnel : la commande `cd` sans argument.

Je ne peux cependant pas sauvegarder d'autre répertoires « favoris » pour pouvoir m'y déplacer facilement.

3 « Système de favoris » : Fonctionnement minimal

L'objectif de ce mini projet est de créer un petit système de « favoris » pour vos répertoires importants.

Votre script devra fournir 4 fonctions :

- S (Save) pour sauvegarder un nouveau favori. Cette fonction prend un argument (une chaîne sans espace) et ajoute le répertoire courant dans votre liste de favoris.
- C (Change) pour se déplacer dans un répertoire favori. Cette fonction prend un argument (une chaîne sans espace) et cherche dans vos favoris. Si le favori existe, la fonction change le répertoire de travail, sinon, elle ne fait rien.
- R (Remove) pour supprimer un favori de la liste.
- L (List) pour afficher la liste de tous les favoris.

Exemple d'utilisation :

Mini Projet

```
$ source favoris.sh
$ pwd
/home/hrenard/LP_DAM/Ecosystemes/MiniProjet/
$ S LP_DAM_MiniProjet
Le répertoire /home/hrenard/LP_DAM/Ecosystemes/MiniProjet/ est sauvegardé dans vos favoris.
-> raccourci : LP_DAM_MiniProjet
$ cd
$ pwd
/home/hrenard/
$ C LP_DAM_MiniProjet
Vous êtes maintenant dans le répertoire /home/hrenard/LP_DAM/Ecosystemes/MiniProjet/
$ pwd
/home/hrenard/LP_DAM/Ecosystemes/MiniProjet/
$ R LP_DAM_MiniProjet
Le favori 'LP_DAM_MiniProjet' est supprimé de votre liste.
$ C LP_DAM_MiniProjet
Le favori 'LP_DAM_MiniProjet' n'existe pas.
```

3.1 Stockage des favoris

Votre script utilisera un fichier (caché) `.favoris_bash` qui sera créé dans votre dossier personnel. Ce fichier contiendra une ligne par favori. Chaque ligne contiendra

- le nom du favori (une chaîne sans espace) ;
- la chaîne de caractères ``->`` ;
- le chemin absolu du répertoire correspondant.

Remarque :

Pour simplifier la gestion de ce fichier, il est conseillé de le définir dans une variable au début de votre script :

```
FAV=$HOME/.favoris_bash
```

Note : la variable `HOME` contient le chemin absolu vers votre dossier personnel

3.2 Manipulation du fichier

Pour manipuler le fichier de favoris, il faudra utiliser les choses suivantes.

- La commande `grep` qui permet d'afficher les lignes contenant une chaîne de caractères donnée. Remarque, si on veut imposer que la chaîne cherchée se trouve en début de ligne, on peut la préfixer par le symbole `^` :
 - `grep "abc"` donne toutes les lignes qui contiennent `abc` ;
 - `grep "^abc"` donne toutes les lignes qui commencent par `abc`.
- Les redirections `CMD > FICHIER` et `CMD >> FICHIER` qui permettent de
 - remplacer le contenu de `FICHIER` par le résultat d'une commande ;

Mini Projet

-
- de rajouter le résultat d'une commande à la fin de FICHIER.

Remarque :

Attention, il n'est en général pas possible de rediriger le résultat d'une commande sur un fichier dans le même fichier. Par exemple,

```
$ grep "CHAINE" fichier > fichier
```

n'aura pas l'effet attendu car la redirection > fichier supprime le contenu avant que la commande grep "CHAINE" fichier ne commence.

Pour faire ceci, il faut donc rediriger la commande dans un autre fichier temporaire, et remplacer ensuite le fichier d'origine par ce fichier temporaire.

4 Pour aller plus loin !

Si le mini-projet vous semble trop facile, vous pouvez éventuellement :

- Autoriser les noms de favori incomplets : C MAM3 utilisera le favori MAM3-Covid s'il existe. La commande C devra donc chercher les favoris « compatibles »
 - s'il n'y en a qu'un seul, elle l'utilise ;
 - s'il y a un favori « exact », elle l'utilise ;
 - sinon, elle affiche un message et ne fait rien.

Bon mini-projet !