TP4-IUP1

Les variables d'environnement (Unix externe)

January 12, 2009

1 Donner une
valeur a une
variable, et
l'exporter

Les commandes importantes :

- set
- echo \$nom_de_variable
- export nom_de_variable
- setenv
- csh
- sh

↳ Exercice

- Exécutez les commandes suivantes sous Unix, et interprétez les résultats que vous avez obtenus.

BOURNE SHELL

```
>sh
>a=truc
>echo $a
>sh
>echo $a
>exit
>echo $a
>export a
>sh
>echo $a
>exit
>exit
```

C-SHELL

```
>csh
>set a=truc
>echo $a
>csh
>echo $a
>exit
>setenv b machin
>echo $b
>csh
>echo $b
>sh
```

- Recherchez dans les pages **man** ce que fait la commande **read**, puis donnez une méthode pour créer rapidement trois variables **nom prenom adresse**.

2 Afficher correctement une variable

↳ Exercice

Exécutez les lignes suivantes (en bourn shell) et interprétez les résultats que vous avez obtenus. Faites attention à bien distinguer les guillemets des quotes et des backquotes.

a

```
>x=ab
>echo $x
>echo $xcd
>echo ${x}cd
```

b

```
>pwd
>a='pwd'
>echo $a
>a='pwd'
>echo $a
>cd ..
>echo $a
>pwd
```

c

```
>a=truc
>x='machin$a'
>y='machin$a'
>echo $x
>echo $y
```

3 Les variables prédéfinis d'environnement ou

La commande importante

- env

Les variables importantes

- PATH
- PWD

Un certain nombre de variables sont prédéfinies lorsque l'on se connecte à une machine. Ces variables peuvent être visualisées à l'aide de la commande `env` (la liste de toutes les variables définies à un moment donné dans un shell s'obtient en utilisant la commande `set` sans arguments). Ces variables sont définies pour tout les programmes. Elles sont dites *exportables*.

↳ Exercice

Un fichier de script est un fichier qui contient une liste de commandes, et qui est exécutable. Le début d'un fichier de script commence par le shell que l'on utilise. Créez dans votre répertoire personnel un répertoire `bin`, puis créez dans ce répertoire le fichier de script qui contient les deux lignes suivantes :

```
#!/bin/sh
ls -l
```

Ce script commence par charger un bourne shell, puis exécute la commande `ls -l`. Pour pouvoir l'exécuter donnez lui la permission `-rwxr-x---`. Et enfin

appelez le.

Pour exécuter une commande qui n'est pas interne, le shell recherche l'exécutable de cette commande dans les répertoires contenus dans la variable d'environnement `PATH`.

↳ **Exercice**

Vérifiez que vous possédez bien dans votre variable d'environnement `PATH` le répertoire courant `.`. Puis créez une commande vous permettant d'ajouter le répertoire courant à votre variable d'environnement `PATH`. Enfin changez de répertoire, et vérifiez que vous pouvez toujours exécuter le script que vous avez créé précédemment.

↳ **Exercice**

Les variables d'environnement `PS1` en BOURNE SHELL, et `prompt` en C SHELL permettent de changer l'allure du prompt. Recherchez dans la documentation, les différentes manières de spécifier un prompt sous bash (regardez de préférence dans les pages info), et sur la méthode pour spécifier un prompt sous `cs`h (utilisez les pages man, mais vérifiez que `cs`h n'est pas un lien symbolique sur une autre commande de shell afin de ne pas vous tromper de page man). Et reformez votre prompt, de manière à ce que celui-ci vous affiche dans l'ordre :

l'heure, la machine sur laquelle vous travaillez, votre nom, et le répertoire dans lequel vous travaillez. Par exemple :

```
01:07pm linux@paranth: /courIUP/TP/TP4 >  
01:14pm linux@paranth: /courIUP/TP/TP4 >
```

Améliorez ce prompt pour qu'il soit souligné.