

Travaux Pratiques

Initiation à la programmation avec le shell Bash

Feuille n.2

Copyright (C) 2014 Jean-Vincent Loddo
Licence Creative Commons Paternité - Partage à l'Identique 3.0 non transposé.

1 Retour sur la conditionnelle

1.1 if-then-elif-else

Utilisez la construction *if-then-elif-else* généralisée de Bash (ou *branchement multiple*, voir `help if`) et la commande `test` (voir `help test`) pour rendre le service suivant : demander à l'utilisateur deux noms de fichiers (avec deux appels de `zenity --file-selection`) et lui indiquer (par `zenity --info`) si le premier contient plus de lignes que le second, ou le contraire, ou bien si les deux ont le même nombre de lignes (il y a donc trois cas possibles...). Pour compter le nombre de lignes d'un fichier, par exemple `foo.txt`, il suffit d'utiliser la commande `wc` (acronyme de word count) en redirigeant son entrée standard sur le fichier en question :

```
wc -l <foo.txt
```

1.2 case

Utilisez la construction *case* de Bash (*filtrage de motifs*, voir `help case`) pour rendre le service suivant : **traduire** le jour de la semaine courant (dimanche, lundi, mardi, ...), fourni par le sous-traitant `date +%A` (essayez), **en latin** (dies Solis, dies Lunae, dies Martis, dies Mercurii, dies Jovis, dies Veneris, dies Saturni). Donnez la précieuse information à l'utilisateur par une fenêtre `zenity --info`.

Pour **tester** la traduction avec d'autres jours de la semaine que le courant, modifiez temporairement l'appel de la commande `date`, par exemple de cette façon :

```
date -d --yesterday +%A
date -d --tomorrow +%A
date -d 2016/11/30 +%A
```

2 Boucles for

2.1 Boucles for de type “foreach”

Rappel de syntaxe :

```
for NOM in MOTS...; do COMMANDES; done
```

Service à rendre : demander à l'utilisateur (`zenity --entry`) de préciser une **extension** de fichier image (`jpg`, `gif`, `bmp`, `pict`, `tif`), de préciser aussi un **répertoire** de travail (`zenity --file-selection`) et convertir ensuite au format `png` (avec `convert`) tous les fichiers de **ce** répertoire avec **cette** extension.

Pour **tester** le programme, créez un répertoire de test et sauvegardez à l'intérieur quelques images téléchargeables sur Internet.

2.2 Boucles for “arithmétiques”

Rappel de syntaxe :

```
for ((EXPR1; EXPR2; EXPR3 )); do COMMANDES; done
```

où les expressions sont admissibles pour la commande `let` (voir `help let`).

Exemple (calcul de 2^5) :

```
N=1; for ((X=1; X<=5; X=X+1)); do N=$((N*2)); done
```

Service à rendre : demander à l'utilisateur deux entiers X et N (par deux appels à `zenity --scale`), calculer la puissance X^N , que vous savez être égale à $X \times X \times \dots \times X$ (N fois), et informez l'utilisateur du résultat par une fenêtre de dialogue (`zenity --info`).

Rappel : pour manipuler les nombres entiers en Bash, vous pouvez soit utiliser l'opérateur `=$(())`, soit le `let` (voir `help let`). Par exemple, pour incrémenter une variable Y , vous avez les deux possibilités suivantes :

```
Y=$((Y+1))
let Y=Y+1
```

3 Boucles while (avec condition de sortie)

3.1 Traduction for arithmétique → while

Transformez la boucle `for` arithmétique de l'exercice 2.2 (puissance X^N) en boucle `while` (voir transparents du cours magistral). Vérifiez que le programme ainsi obtenu soit équivalent à l'original.

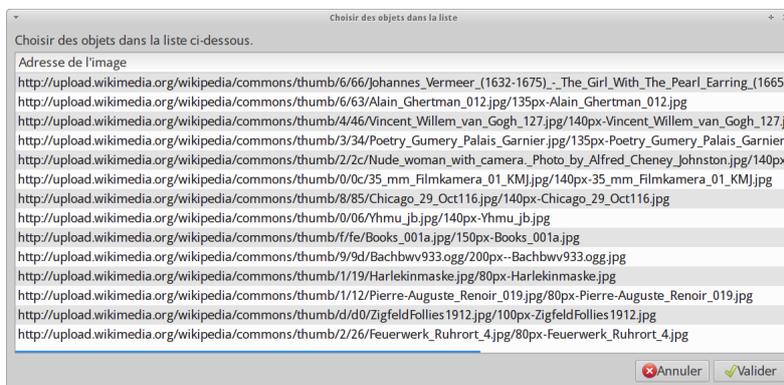
3.2 Télécharger et afficher des images en boucle

La commande (pipeline) suivante :

```
lynx --dump -image_links "http://fr.wikipedia.org/wiki/Portail:Arts" \
| grep -o "http://upload.wikimedia.*jpg" \
| zenity --list --column "Adresse de l'image"
```

permet d'obtenir une liste d'adresses HTTP d'images à partir de `http://fr.wikipedia.org/wiki/Portail:Arts`. L'effet du pipeline est celui de fournir une liste de choix pour `zenity`. (**Commentaire :** `lynx` est un navigateur en modalité texte; la commande `grep` permet de sélectionner les lignes contenant une adresse d'image JPEG parmi les lignes de la sortie de `lynx`; enfin, `zenity --list` lit la liste des options depuis son entrée standard et la présente à l'utilisateur).

L'utilisateur pourra donc choisir l'adresse qui l'intéresse :



D'autre part, une commande du genre :

```
while zenity --question --text="Continuer?"; do
...
done
```

répète un comportement (...) tant que l'utilisateur répond positivement. **Assembler** ces deux constructions pour afficher l'image choisie par l'utilisateur (utiliser `firefox` ou `chrome`) tant que celui-ci voudra continuer.

4 Paramétrage

Paramétrez le programme précédent de façon à pouvoir y faire appel avec un "portail wikipedia" particulier, pas forcément `Arts`. Pour tester le programme, qu'on appellera par exemple `affiche_images_portail.sh`, essayez quelques uns des portails accessible à partir de `http://fr.wikipedia.org/wiki/Portail:Accueil`. Exemple d'appel possible :

```
affiche_images_portail.sh Animaux_de_compagnie
```