

Introduction au script shell

C. Beaume

Master 2 Professionnel MSME

Rappels

Les langages du Shell

sh	premier shell
bash	extension du sh
csh	proche du C
tcsh	extension du csh
ksh	intermédiaire entre bash et csh
zsh	mélange de bash, ksh et tcsh

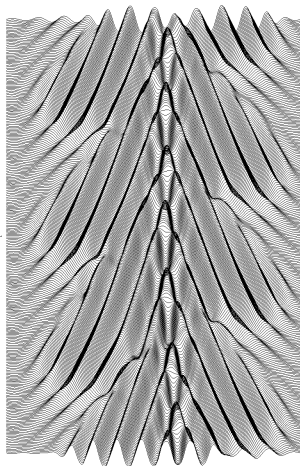
⇒ Utilisation d'internet quasi-indispensable !

Généralités

- `#!/bin/bash` commence toujours pour spécifier le langage.
- `#` sert à introduire un commentaire.
- `echo` sert à écrire.
- Utiliser `bc` (basic calculator) pour calculer.
- Beaucoup de commandes sont identiques dans tous les langages. Souvent, seule la syntaxe change.
- Pour rendre un script exécutable, il faut le “`chmoder`”.

Un exemple en bash - 1

Comment tracer cette figure malgré les contraintes liées au nombre de fichiers et à leur syntaxe ?



Un exemple en bash - 2

toto.sh

```
#!/bin/bash

#-----
# Variables
#-----

current=1
ndata=101
xlength=10
ylength='echo "scale=2;($ndata / 100)" | bc'

nline='wc -l SOL.$current.dat | cut -d" "
-f1'
ltoplot='echo "$nline - 2" | bc'

#-----
# Generating gnuplot script
#-----

echo "set term postscript enhanced eps
\" Times-Roman\" 24 size $xlength
cm,$ylength cm" >> wmid.plt
echo "set output \"wmid.eps\"" >> wmid.plt

wplot="plot"

for i in $(seq 1 $ndata);
do

wmax='awk -v l=$nline 'NR==1{ print $2 }'
SOL.$current.dat'
echo 'le max est $wmax'
wplot1=" 'SOL.$current.dat' u 1:2
ev ::0::$ltoplot w l"
wplot=$wplot$wplot1
current='echo "$current + 1" | bc'

if [ $i != $ndata ];
then
wplot1=","
wplot=$wplot$wplot1
fi

done

echo $wplot >> wmid.plt

gnuplot wmid.plt
gv wmid.eps
```

Exercice 7

Écrire un script “toto.sh” dans votre répertoire racine qui :

- crée un dossier “script”
- copie tous les fichiers du dossier “environnement” dont le nom commence par “exercice” dans “script”
- remplace “exercice” dans le nom des fichiers contenus dans “script” par “texte”
- affiche à l’écran une phrase indiquant le nombre de fichiers copiés
- archive et comprime l’ensemble du dossier “script” en “script.tar.gz”

Exercice 8

Écrire un script “titi.sh” dans le votre répertoire racine qui :

- requiert un argument à l'exécution
- teste si cet argument est un fichier ou un dossier qui existe
- affiche un message pour dire si l'argument est un fichier, un dossier ou n'existe pas

Run control

`.bashrc`, `.cshrc`, etc...

sert à configurer la session

`alias` : crée un raccourci

`source` : exécute un script d'installation

`export` : crée un lien vers une librairie