

Solutions to **UNIX / Linux & Shell** **exercises**

Katja Winger, UQAM
UQAM, June 2013

Part 1

```
cd /skynet1_rech2/cordex/NorthAmerica_0.44deg_ERA40-Int/Samples
```

1) What is the **size of this directory** in GB?

```
du -s
```

2) **How many** first level sub-directories are there in this directory?

```
ls -1 | wc -l
```

3) **How many** January sub-directories are there in this directory?

```
ls -1d *01 | wc -l
```

4) **List the largest** of all the **files inside** the February directories.

```
ls -lS 02/ | head -1
```

5) **Create a directory** called

```
/skynet1_exec2/${USER}/Training/CRCM5
```

Inside this directory **create a link** called 'NorthAmerica_Samples' pointing to the directory mentioned in 1)

```
mkdir -p /skynet1_exec2/${USER}/Training/CRCM5
```

```
cd /skynet1_exec2/${USER}/Training/CRCM5
```

```
ln -s /skynet1_rech2/cordex/NorthAmerica_0.44deg_ERA40-Int/Samples NorthAmerica_Samples
```

6) **Save** in a file all your previous commands that you typed in the current window into a **file** called: ~/UNIX_Training/Commands.txt

```
mkdir ~/UNIX_Training
```

```
history > ~/UNIX_Training/Commands.txt
```

7) Go into your directory
/skynet1_exec2/\${USER}/Training/CRCM5/NorthAmerica_Samples
Verify where you are with

pwd

And where you really are with

true_path [-n] .

8) Create an alias in your **~/.profile.d/.interactive_profile** for
true_path -n

I called mine 'tp'

alias tp='true_path -n'

9) Check that you got your alias with:

alias tp

10) Log out of skynet1 and back in.
Do you have your 'tp' alias now?

Look for examples in my scripts underneath my directory ~winger/Scripts

11) Find the script (extension '.scr') underneath my directory ~winger/Scripts which
contains a line with: '\${exp}', '-s', and the whole word 'i'

find ~winger/Scripts -name '*.scr' -exec grep '\${exp}' {} \; | grep -- -s | grep -w i

Script_A.scr

Solution suggestion 1 for points 1) - 3)

```
#!/bin/ksh

startyear=1990
endyear=2100

months="01 02 03 04 05 06 07 08 09 10 11 12"

# =====

# 1) Loop over years
year=$startyear
while (( year <= endyear )) ; do

# 2) Loop over months
for month in $months ; do

# 3) Determine number of days per month
case ${month} in
  04|06|09|11) days=30 ;;
  02) if (( year % 400 == 0 )) ; then
        days=29
      elif (( year % 100 == 0 )) ; then
        days=28
      elif (( year % 4 == 0 )) ; then
        days=29
      else
        days=28
      fi ;;
  *) days=31 ;;
esac

  echo "$year $month $days"

done
year=$(( year + 1 ))
done
```

Comment:

It is not necessary to precede variables by '\$' when they stand inside double round parenthesis '((...))'. Since only be arithmetic expressions can be inside '((...))' any string must be a variable.

Script_A.scr

Solution suggestion 2 for points 1) – 3)

(different if-statement to determine leap years)

```
#!/bin/ksh

startyear=1990
endyear=2100

months="01 02 03 04 05 06 07 08 09 10 11 12"

# =====

# 1) Loop over years
year=$startyear
while (( year <= endyear )) ; do

# 2) Loop over months
for month in $months ; do

# 3) Determine number of days per month
case ${month} in
04|06|09|11) days=30 ;;
02) if (( year % 4 == 0 )) && \
      ( (( year % 100 != 0 )) || (( year % 400 == 0 )) ) ; then
      days=29
    else
      days=28
    fi ;;
*) days=31 ;;
esac

echo "$year $month $days"

done
year=$(( year + 1 ))
done
```

Comment:

A '\' as the very last character of a line means that the following line is a continuation of the line with the '\' at the end.