Linux Fundamental-Week4

Hou, Jue

[jue.hou@helsinki.fi](mailto:jue.hou@helsinki.fi)

Exercise 1

Save “the cat” in the shortcat.txt

The script for short cat or long cat:

#!/bin/bash

if [[ $1 =~ ^[0-9]+$ ]]

then

if [ $1 -lt 0 ]

then

echo 'Invalid Input'

exit 1

fi

if [ $1 -gt 1 ]

then echo 'Longcat' $1

else echo 'Shortcat'

fi

head -9 shortcat.txt

for ((i=0 ;i<$1 ;i++))

do

echo ' | |'

done

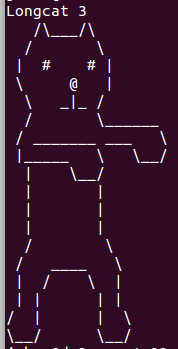
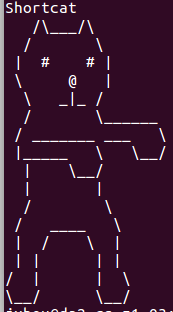
tail -6 shortcat.txt

else

echo 'Invalid Input'

exit 1

fi



Exercise 2

The sript for creating n sets of random number:

#!/bin/bash

echo $RANDOM $RANDOM>random-data.txt

for((i=1;i<$1;i++))

do

echo $RANDOM $RANDOM>>random-data.txt

done

By execute the following command:

./create-random-data.sh 100

echo 'set terminal dumb;plot "random-data.txt" using 1:2;' | gnuplot



Exercise 3:

First, extract the maximum and minium temperature from the text files for November 2011 into a text file by using the following script:

#!/bin/bash

cat /dev/null > temp-max-min.txt

startdate=$(date -d"2011-11-01 00:00:00" +%s)

enddate=$(date -d"2011-12-01 00:00:00" +%s)

for ((i=startdate;i<enddate;i=i+86400))

do

date=$(date -d @$i +%Y.%m.%d)

max\_value=0

curr\_value=0

min\_value=100

for arg in $(find $1/$date -name 'hp-temps.txt')

do

if [ -f $arg ]

then

curr\_value=$(grep 'PROCESSOR\_ZONE' $arg |rev|cut -c 13-21|rev|cut -d 'C' -f 1)

fi

if [ $max\_value -lt $curr\_value ]

then

max\_value=$(grep 'PROCESSOR\_ZONE' $arg |rev|cut -c 13-21|rev|cut -d 'C' -f 1)

fi

if [ $min\_value -gt $curr\_value ]

then

min\_value=$(grep 'PROCESSOR\_ZONE' $arg |rev|cut -c 13-21|rev|cut -d 'C' -f 1)

fi

done

echo $date $max\_value $min\_value

echo $date $prev\_value $min\_value>>temp-max-min.txt

done

By executing the following command, we can get the eps file for maximum temperature of each day in Novemeber 2011:

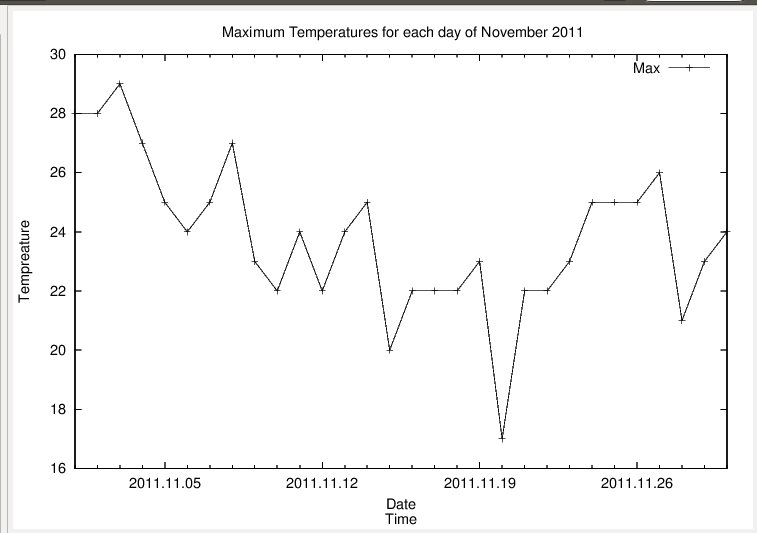
echo 'set terminal postscript eps enhanced;set out "temps-2011-11.eps";

set xdata time;set timefmt "%Y.%m.%d";set format x "%Y.%m.%d";set xlabel "Date\nTime";

set ylabel "Tempreature";set title " Maximum Temperatures for each day of November 2011";

set xrange ["2011.11.01":"2011.11.30"];

plot "temp-max-min.txt" using 1:2 with linespoints title "Max";'|gnuplot



By executing the following command, we can get the eps file for maximum and minium temperature of each day in Novemeber 2011:

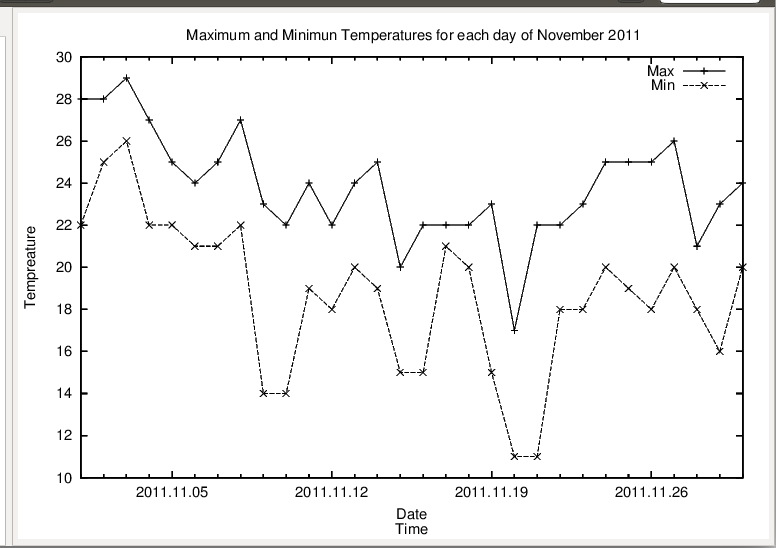
echo 'set terminal postscript eps enhanced;set out "min-max-temps-2011-11.eps";

set xdata time;set timefmt "%Y.%m.%d";set format x "%Y.%m.%d";set xlabel "Date\nTime";

set ylabel "Tempreature";set xrange ["2011.11.01":"2011.11.30"];

set title " Maximum and Minimun Temperatures for each day of November 2011";

plot "temp-max-min.txt" using 1:2 title "Max" with linespoints ,"temp-max-min.txt" using 1:3 title "Min" with linespoints ;'|gnuplot



For generalization, I modified the privious script as followed:

#!/bin/bash

time=${1:0-7:7}

year=${time:0:4}

month=${time:0-2:2}

path=${1%%$time}

touch temp-max-min-$year-$month.txt

startdate=$(date -d"$year-$month-01 00:00:00" +%s)

enddate=$(date -d"$year-$((month+1))-01 00:00:00" +%s)

for ((i=startdate;i<enddate;i=i+86400))

do

date=$(date -d @$i +%Y/%m/%d)

max\_value=0

curr\_value=0

min\_value=100

if [ ! -e $path$date ]

then

continue

fi

for arg in $(find $path$date -name 'hp-temps.txt')

do

if [ -f $arg ]

then

curr\_value=$(grep 'PROCESSOR\_ZONE' $arg |rev|cut -c 13-21|rev|cut -d 'C' -f 1)

fi

if [ $max\_value -lt $curr\_value ]

then

max\_value=$(grep 'PROCESSOR\_ZONE' $arg |rev|cut -c 13-21|rev|cut -d 'C' -f 1)

fi

if [ $min\_value -gt $curr\_value ]

then

min\_value=$(grep 'PROCESSOR\_ZONE' $arg |rev|cut -c 13-21|rev|cut -d 'C' -f 1)

fi

done

echo $date $max\_value $min\_value

echo $date $max\_value $min\_value >>temp-max-min-$year-$month.txt

done

echo 'set terminal postscript eps enhanced;

set out "min-max-temps-'$year-$month'.eps";

set xdata time;set timefmt "%Y/%m/%d";

set format x "%Y/%m/%d";set xlabel "Date\nTime";

set ylabel "Tempreature";

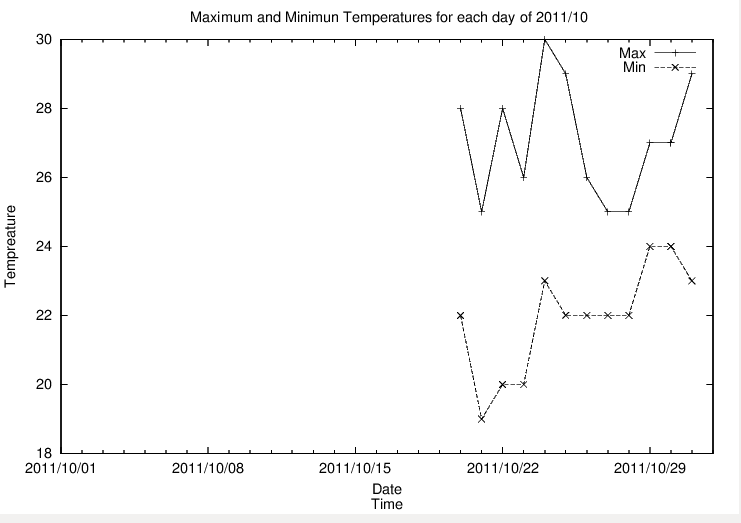
set title " Maximum and Minimun Temperatures for each day of '$year/$month'";

set xrange ["'$year/$month'/01":"'$year/$((month+1))'/01"];

plot "temp-max-min-'$year-$month'.txt" using 1:2 title "Max" with linespoints ,"temp-max-min-'$year-$month'.txt" using 1:3 title "Min" with linespoints ;'|gnuplot

rm ./temp-max-min-$year-$month.txt

The parameter of this script has to be the document name which is represent for a specific month, for example “./script.sh /$YOURPATH /lost24/monitor/2011/10”



Exercise 4

The script which can take a few options by using getopts and switch:

Due to the length of this script, I will explain my solution briefly first and leave the script in the end.

The basic idea for my script is to extract max,min or both temperature of each in a month to a file according to the argument into a temp file. Since the exercise mentioned that there has to be two different type of output, I let the plot to be generated at the end of the script so that a variable can be applied to change the type of the output. And then, clean up the temp file and that’s it.

The script wrap the privious several scripts:

#!/bin/bash

cat /dev/null > temp.txt

type1=''

title=''

output=''

outputfile=''

while getopts ":ab: :aw: :ac: :w: :b: :c: :h" opt;

do

case $opt in

c)

time=${OPTARG:0-7:7}

year=${time:0:4}

month=${time:0-2:2}

path=${OPTARG%%$time}

echo $OPTARG,$time,$year,$month

startdate=$(date -d"$year-$month-01 00:00:00" +%s)

enddate=$(date -d"$year-$((month+1))-01 00:00:00" +%s)

for ((i=startdate;i<enddate;i=i+86400))

do

date=$(date -d @$i +%Y/%m/%d)

curr\_value=0

min\_value=100

if [ ! -e $path$date ]

then

continue

fi

for arg in $(find $path$date -name 'hp-temps.txt')

do

if [ -f $arg ]

then

curr\_value=$(grep 'PROCESSOR\_ZONE' $arg |rev|cut -c 13-21|rev|cut -d 'C' -f 1)

fi

if [ $min\_value -gt $curr\_value ]

then

min\_value=$(grep 'PROCESSOR\_ZONE' $arg |rev|cut -c 13-21|rev|cut -d 'C' -f 1)

fi

done

echo $date $min\_value >>temp.txt

done

if [ $output = '' ]

then

output='set terminal postscript eps enhanced;'

outputfile='set out "min-temps-'$year-$month'.eps";'

fi

type1='min'

title='Minimun'

;;

w)

time=${OPTARG:0-7:7}

year=${time:0:4}

month=${time:0-2:2}

path=${OPTARG%%$time}

startdate=$(date -d"$year-$month-01 00:00:00" +%s)

enddate=$(date -d"$year-$((month+1))-01 00:00:00" +%s)

for ((i=startdate;i<enddate;i=i+86400))

do

date=$(date -d @$i +%Y/%m/%d)

max\_value=0

curr\_value=0

if [ ! -e $path$date ]

then

continue

fi

for arg in $(find $path$date -name 'hp-temps.txt')

do

if [ -f $arg ]

then

curr\_value=$(grep 'PROCESSOR\_ZONE' $arg |rev|cut -c 13-21|rev|cut -d 'C' -f 1)

fi

if [ $max\_value -lt $curr\_value ]

then

max\_value=$(grep 'PROCESSOR\_ZONE' $arg |rev|cut -c 13-21|rev|cut -d 'C' -f 1)

fi

done

echo $date $max\_value

echo $date $max\_value >>temp.txt

done

if [ $output = '' ]

then

output='set terminal postscript eps enhanced;'

outputfile='set out "max-temps-'$year-$month'.eps";'

fi

type1='max'

title='Maximun'

;;

b)

time=${OPTARG:0-7:7}

year=${time:0:4}

month=${time:0-2:2}

path=${OPTARG%%$time}

startdate=$(date -d"$year-$month-01 00:00:00" +%s)

enddate=$(date -d"$year-$((month+1))-01 00:00:00" +%s)

for ((i=startdate;i<enddate;i=i+86400))

do

date=$(date -d @$i +%Y/%m/%d)

max\_value=0

curr\_value=0

min\_value=100

if [ ! -e $path$date ]

then

continue

fi

for arg in $(find $path$date -name 'hp-temps.txt')

do

if [ -f $arg ]

then

curr\_value=$(grep 'PROCESSOR\_ZONE' $arg |rev|cut -c 13-21|rev|cut -d 'C' -f 1)

fi

if [ $max\_value -lt $curr\_value ]

then

max\_value=$(grep 'PROCESSOR\_ZONE' $arg |rev|cut -c 13-21|rev|cut -d 'C' -f 1)

fi

if [ $min\_value -gt $curr\_value ]

then

min\_value=$(grep 'PROCESSOR\_ZONE' $arg |rev|cut -c 13-21|rev|cut -d 'C' -f 1)

fi

done

echo $date $max\_value $min\_value

echo $date $max\_value $min\_value >>temp.txt

done

if [ $output = '' ]

then

output='set terminal postscript eps enhanced;'

outputfile='set out "min-max-temps-'$year-$month'.eps";'

fi

;;

a)

output='set terminal dumb;'

outputfile=''

;;

h)

echo "Usage [OPTION] [ARGUMENT]

The argument should be a directory.

-c: for coldest temperatures

-w: for warmest temperatures

-b: for both temperatures

-ac: to produce ASCII output instead of eps(coldest temperatures)

-aw: to produce ASCII output instead of eps(warmest temperatures)

-ab: to produce ASCII output instead of eps(both temperatures)

-h: for help " >&2

;;

\?)

echo "Invalid option: -$OPTARG" >&2

;;

:)

echo "Parameter required: -$OPTARG" >&2

;;

esac

done

echo $output$outputfile

if [ ! $1 = '-h' ]

then

if [[ ! ${1%% \*} == \*b\* ]]

then

echo $output$outputfile'set xdata time;set timefmt "%Y/%m/%d";

set format x "%Y/%m/%d";set xlabel "Date\nTime";

set ylabel "Tempreature";

set title "'$title' Temperatures for each day of '$year/$month'";

set xrange ["'$year/$month'/01":"'$year/$((month+1))'/01"];

plot "temp.txt" using 1:2 title "'$type1'" with linespoints ;'|gnuplot

else

echo $output$outputfile'set xdata time;set timefmt "%Y/%m/%d";

set format x "%Y/%m/%d";set xlabel "Date\nTime";

set ylabel "Tempreature";

set title " Maximum and Minimun Temperatures for each day of '$year/$month'";

set xrange ["'$year/$month'/01":"'$year/$((month+1))'/01"];

plot "temp.txt" using 1:2 title "Max" with linespoints ,"temp.txt" using 1:3 title "Min" with linespoints ;'|gnuplot

fi

rm temp.txt

fi

