

CTIS 166::Information Technologies::SPRING 2014-2015
Computer Technology and Information Systems, Bilkent University
Solution of FIRST Midterm Examination

Instructors: Mustafa Akgül and Hamdi Murat Yıldırım

A / Duration: 110' / 23.03.2015/

Section Number:1,2.....

Closed Book, closed note exam.

You are required to write down commands with necessary arguments and options; and make sure that they work. Your script and output should match.

Give the best result that you can give!

**Over 100 points is bonus. **

Unless otherwise stated for question k, your answers as command must be in k.sh and output should in k.txt both should be in Answers directory. }

You can write down short scripts on exam paper, but you must have the output file and it should match.

Prelude: before solving questions you should:

- **let NAME be your FirstLast name as ascii (MAkgul, ASOzgur, LMessi, LionelMessi)**
- **create NAME and NAME/Answers directories \verb+mkdir -p ~/NAME/Answers+**
- **script NAME/Answers/NAME.Log**
- **touch NAME/Answers/Your-Full-Name**
- **download the data file and unzip it in NAME Directory, use unzip -X -K LabM1.zip**
- **mkdir -p ~/NAME/Answers/Dir{1,2,3,4,5,6,7,8,9,10,11,12,13}**

**When you finish} (that is when exam ends), you will zip NAME directory with command
cd ; zip -r NAME NAME ; upload NAME.zip using browser into lab4t server**

enter the LabM directory

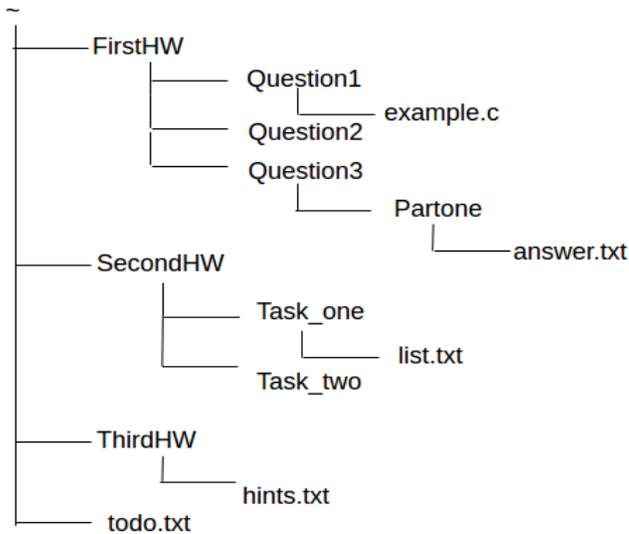
All commands are executed in LabM, unless otherwise stated. You can move temporarily in other directory using subshell, e.g (cd ../Answers/dirx; tar xf ../xyz.tar). Also alternatif solutions are given.

1) (7 points)

Create the directory structure shown on the below in ~/NAME/LabM

(Hint: In order to create files, you can use the command *touch*).

Assume that your present working directory is ~/NAME/LabM and you run all commands while you are in that directory. You are required to write down all related command or commands necessary to create this directory structure.



You could combine some of these commands, but easiest to understand might be the following

```
mkdir -p FirstHW/Question{1,2,3}/Partone}
mkdir -p SecondHW/Task_{one,two}
mkdir ThirdHW
touch FirstHW/Question{1/example.c,3/Partone/answers.txt}
touch SecondHW/Task_one/list.txt
touch {ThirdHW/hints.txt,todo.txt}
```

2) (2 points) How can you display the contents of the directory *FirstHW*. (and write in *../Answers/2.txt*)

```
ls -lR > ../Answers/2.txt
```

3) copy everything in LabM keeping properties in Dir1

```
cp -Rp . ../Answers/Dir1
```

4) copy everything in LabM into Dir2 using rsync

```
rsync -av . ../Answers/Dir2
```

5) copy everything in LabM into Dir3 using tar

```
tar cf - . | tar xf - -C ../Answers/Dir3
```

```
tar cf - . | (cd ../Answers/Dir3; tar xf -)
```

```
tar cf ../Answers/Dir3.tar; (cd ../Answers/Dir3; tar xf ../Dir3.tar)
```

6) copy everything in LabM into Dir4 using cpio

```
find . | cpio -pd ../Answers/Dir4
```

7) **(2 points)** Put the copy the file *list.txt* inside the directory **Question2**.

```
cp SecondHW/Task_one/list.txt FirstHW/Question2
```

8) **(2 points)** Move the directory **Question3** inside the directory **ThirdHW**

```
mv FirstHW/Question3 ThirdHW
```

9) **(2 points)** Remove/Delete the non-empty directory **Question3** .

```
rm -r ThirdHW/Question3
```

10) **(2 points)** copy *input*.txt* into *Dir5* and compress *input1.txt* fast with *gzip*, compress *input2.txt* with *bzip2* most , and compress *input3.txt* with *xz* in *Dir5*

```
cp input*.txt ../Answers/Dir5
(cd ../Answers/Dir5; gzip -1 input1.txt; bzip2 -9 input2.txt; xz input3.txt )
```

11) **(2 points)** copy *Dene** into *Dir6* and uncompress them in *Dir6*

```
cp Dene* ../Answers/Dir6
(cd ../Answers/Dir6; unxz Dene1.txt.xz, gunzip Dene2.txt.gz; bunzip2 Dene3.txt.bz2 )
```

Note that: *unxz = xz -d* *gunzip= gzip -d* *bunzip2 = bzip2 -d*

12) **(2 points)** compress all files inside the directory **FirstHW**

```
only gzip is recursively
gzip -r FirstHW
```

13) **(2 points)** Create an archive file *My.tar* containing the directories *ABC*, and **SecondHW**

```
tar cvf ../Answers/My.tar ABC SecondHW
```

14) **(2 points)** extract contents of directory *ABC* of *my.tar* into *Dir7*

```
tar xvf ../Answers/My.tar -C ../Answers/Dir7 ABC
```

```
(cd ../Answers/Dir7; tar xvf ../My.tar ABC )
```

15) combine all **.txt* files on surface into *15.txt* in **Answers**

```
cat *.txt > ../Answers/15.txt
```

more advance one:

```
find . -maxdepth 1 -type f -name "*.txt" | xargs cat > ../15.txt
```

16) find all **.txt* files in **LabM** write file names in *16.txt* in **Answers**

```
find . -type f -name "*.txt" > ../Answers/16.txt
```

17) find all *.txt files in LabM and put them in TXT.zip in Answers

```
find . -type f -name "*.txt" | xargs zip ../Answers/TXT.zip
```

```
find . -type f -name "*.txt" -exec zip ../Answers/TXT '{}' \;
```

18) find all *.txt files in LabM and put them in TXT.tar in Answers

```
find . -type f -name "*.txt" | xargs tar cvf ../Answers/TXT.tar
```

```
find . -type f -name "*.txt" -exec tar rvf ../answers/TXT.tar '{}' \;
```

19) find all *.txt files which are newer than file time.stamp and place them in Dir8 keeping the directory structure

```
find . -type f -name "*.txt" -newer time.stamp | cpio -pd ../Answers/Dir8
```

```
find . -type f -name "*.txt" -newer time.stamp | xargs tar cf ../Answers/19.tar  
tar xf ../Answers/19.tar -C ../Answers/19.tar
```

```
find . -type f -name "*.txt" -newer time.stamp | xargs zip ../Answers/19.zip  
(cd ../Answers/Dir8; unzip ../19.zip)
```

20) create a link to directory /etc in Answer

```
ln -s /etc ../Answers
```

21) create a soft link to file /etc/passwd in Answers

```
ln -s /etc/passwd passwd
```

22) create a hard link to file A.txt as B.txt in LabM

```
ln A.txt B.txt
```

23) (4 points) Consider directories /etc /proc /dev /usr under the root directory of the Linux file system. Briefly explain which kinds of files available inside each of these four directions. (You can write in 23.txt)

/etc: mainly configuration files

/proc: snapshot of the system at that time, info about hardware and processes, it also allows reconfiguration certain aspects of system

/dev, defines various devices

/usr programs, libraries and documentation for (end) user related programs/services

24) (3 points) Determine the full pathname of the **apropos** command's binary or executable file in a Linux system? (write the command and the path) also list the related files in the system

```
which apropos : /usr/bin/apropos
```

```
whreis apropos:
```

```
apropos: /usr/bin/apropos /usr/share/man/man1/apropos.1
```

25) (2 points) Given Dene.txt of 103 lines, how would you obtain dene.txt containing lines 21-90 with lines numbers as 25.txt in Answers

```
cat -n Dene.txt | head -90 | tail -70 > ../Answers/25.txt
```

```
cat -n Dene.txt | tail -83 | head -70 > ../Answers/25.txt
```

```
cat -n Dene.txt | head -n -13 | tail -n +23 > ../Answers/25.txt
```

- 26) (2 points) Determine lines containing string fox case insensitive in Dene1.txt.xz without uncompressing Dene1.xz (put result in 26.txt in Answers)

```
xzgrep -i fox Dene1.txt.xz > ../Answers/26.txt
```

- 27) (3 points) How would you create the empty text files **a3.txt a4.txt a5.txt a6.txt b3.txt b4.txt b5.txt b6.txt c3.txt c4.txt c5.txt c6.txt d3.txt d4.txt d5.txt d6.txt e3.txt e4.txt e5.txt e6.txt** using a single command (you are required to use suitable shell metacharacters)? (in Answers)

```
touch ../Answers/{a,b,c,d,f}/{3,4,5,6}.txt
```

- 27x) (3 points) Create directories A B C D each containing X Y Z and each in turn containing d e f in a single command. (in Answers)

```
mkdir -p ../Answers/{A,B,C,D}/{X,Y,Z}/{d,e,f}
```

28)

- a) (3 points) How would you learn the locations of all directories whose filenames do not start with a letter, second character is a digit and then followed by zero or more characters? This command recursively searches the directory **/usr**. (put the list of directories in 28a.txt in Answers)

```
find /usr -type d -name "[^a-zA-Z][0-9]*" > ../Answers/28a.txt
```

- b) (4 points) Determine list of empty directories under **/usr** (put in 28b.txt) and determine number of such directories (put in 28b.TXT in Answers or write that number here)

```
find /usr -type d -empty > ../Answers/28b.txt
```

```
find /usr -type d -empty < wc -l > ../Answers/28b.TXT
```

(You may get a different answer because of different systems)

29)

- a) (2 points) copy **Dene*txt*** into **Dir8** and uncompress them

This is same as question 11 (Dir8 is also used question 9. you may use Dir13)

- b) (2 points) copy all hidden files and hidden directories which are on the surface in **LabM** into **Dir9**

Only files **.alpha** and **.cokgizli** and directory **.Gizli** is on the surface

```
cp -R .alpha .cokgizli .Gizli ../answers/Dir9
```

- 30) (3 points) Write down a command to find recursively all files inside directory **/etc** whose size is greater than 90 KB and copy them into **Dir10**

```
find /etc -type f -size +90k -exec cp '{}' ../Answers/Dir10 \;
```

```
find /etc -type f -size +90k | xargs cp -t ../Answers/10
```

- 31) (3 points) How would you learn the locations of all standard files whose filenames start with a letter and followed by any two characters other than letters between a and t ? This command recursively searches the directory */usr*
put filename sin 31.txt in Answers

```
find /usr -type f -name "[A-Za-z][^a-t][^a-t]" > ../Answers/31.txt
```

- 32) (4 pts) How would determine files type of all files in directory /dev. And list different file types and give an example of each type

just looking at first character of result of "ls -l object"

c means character device e.g. /dev/audio

b block device e.g /dev/sd1

l soft link e.g /dev/dvs → /dev/sr0

s socket e.g /dev/log

- orinaty file

d directory

p named pipe

- 33) (2 points) Determine number of empty lines in AA.txt

```
grep -c ^$ AA.txt > ../Answers/33.txt
```

```
grep ^$ AA.txt | wc -l > ../Answers/33.txt
```

- 34) (3 points) Determine lines in A.txt which contains at least one of elif and ayse

```
egrep "elif|ayse" A.txt > ../Answers/34.txt
```

- 35) (2 points) Determine lines of A.txt which contains both of elif and ayse

```
grep elif A.txt | grep ayse > ../Answers/35.txt
```

- 36) (3 points) Determine lines of A.txt which contains none of elif and ayse

```
egrep -v "elif|ayse" A.txt > ../Answers/36.txt
```

```
grep -v elif A.txt | grep -v ayse > ../Answers/36.txt
```

- 37) (2 points) Print the list of all manual pages whose one-line description part contain the string **disk**.

```
aropos disk > ../Answers/37.txt
```

```
man -k disk > ../Answers/37.txt
```

38) (2 points) Determine all empty files in Dir1 and write name of files in 38.txt in Answers

```
(cd ../Answers/Dir1; find . -type f -empty > ../38.txt )
```

```
find ../Answers/Dir1 -type f -empty > ../Answers/38.txt
```

39) (2 pts) Delete all empty files in Dir1 (do this after question 38)

```
find ../Answers/Dir1 -type f -empty -delete
```

40) (3 pts) Create a directory named Upload in Answers so that everybody (user,group,others) can create files but can not see file names.

```
Mkdir ../Answers/Upload; chmod 0333 ../answers/Upload
```

41) (3 pts) Assume you have an Rsync Server (name will be given in class). Determine list of all visible modules on the Rsync Server

```
rsync -av liste.ctis.bilkent.edu.tr:: > ../Answers/41.txt
```

42) (3 pts) copy contents of public module temp into Dir11

```
rsync -av liste.ctis.bilkent.edu.tr::temp ../Answers/Dir11
```

43) (3 pts) create a file containing your full name with name NAME.txt. Upload it into module gizli, which requires user ctis with passwd ctisxx

```
rsync -av NAME.txt ctis@liste.ctis.bilkent.edu.tr::gizli
```

44) (4 points) Determine list of *.txt files in LabM which are on surface, which contains string net case insensitive (write into 44.txt) and combine these files into 44.TXT

```
grep -il net *.txt > ../Answers/44.txt
```

```
grep -il net *txt | xargs cat > ../Answers/44.TXT
```

45) (3 pts) Determine lines of Dene.txt which contains string elif followed by ayse

```
grep "elif.*ayse" Dene.txt > ../Answers/45.txt
```