**1- Funções:**

‘pwd’ (print working directory): mostra diretório operante atual;

‘cd’ (change directory): alterar o diretório atual do shell

‘ls’: Lista arquivos do diretório;

‘ln’ (link): cria link entre arquivos;

‘df’ (disk free): apresenta quantidade de espaço do disco;

‘w’ (who, uptime, and ps -a.): apresenta quem esta logado na máquina, o que esta fazendo, e os recursos utilizados;

‘who’: Mostra usuários logados no sistema;

‘whoami’("Who am I?"): Apresenta nome do usuário que o evocou;

‘mkdir’ (“make directory.”): cria um diretório;

‘cp’(“copying”): Copia um ou múltiplos arquivos;

‘mv’(“move”): Move arquivos e diretórios;

‘rm’(“remove files”): Remove um arquivos ou diretórios;

‘rmdir’ (“remove directory”): remove diretórios vazios;

‘cat’ (“con**cat**enate”): permite criar um/vários arquivos, ver seu conteúdo, concatena-los e/ou redireciona-los em um terminal ou outro arquivo;

‘head’ : mostra as 10 (dez) primeiras linhas de um arquivo. E também pode mostrar apenas o número pré definido com

‘head –n [número]’;

‘tail’: mesmo que o comando ‘head’, mas ao invés do topo/início do programa, o final dele;

‘more’: usado para mostrar arquivos de texto no prompt;

‘tar’: Cria ou extrai arquivos, muito usado como programa de backup ou compressÃ£o de arquivos

‘chmod’(“change mode”): Muda proteção de arquivo ou diretório e sinalizações;( 1,2,4);

‘chown’(“change owner”): Muda dono de um arquivo ou diretório;

‘chgrp’ (“change group”): Muda o usuário e/ou grupo de um arquivo/diretório/link simbólico;

‘file’: reconhece tipo de data de um arquivo;

‘clear’: limpa o conteúdo de um terminal;

‘man’: manual descritivo. Usado para saber mais sobre uma lista de comandos ou sobre um determinado comando;

‘wc’ (“word count”): Lê tanto letras e palavras quanto arquivos e gera uma ou mais estatísticas ;

‘ps’(“process status”): gera informação sobre processo em execução atualmente, incluindo PIDs;

‘top’: apresenta lista de uso da memória do Linux e todos os processos em execução. Similar ao ‘task manager’ do Windows;

‘kill’: finaliza um processo determinado pelo usuário;

‘shutdown’: programa desligamento/reinicialização do sistema operacional;

**2- código:** if test –f”$1”

then

echo “file”

elif test –f”$1”

then

echo “directory”

else  
 echo “invalid”

fi

**3- código:**

echo "Enter Celsius temperature"

read CELS

FAHREN=$(((($CELS\*9)+160)/5))

echo "$CELS C is $FAHREN F"

**4- código:**

read NUM

if test "$NUM" -lt 0

then

echo "NEGATIVO"

elif test "$NUM" -gt 0

then

echo "POSITIVO"

else

echo "NULO"

fi

**5- código:**

i=1

max=0

echo "Enter Numbers:"

while [ $i -le 3 ]

do

read num

if [ $i -eq 1 ] #set first number as max

then

max=$num

else #from number 2 update max if the num > max.

if [ $num -gt $max ]

then

max=$num

fi

fi

i=$((i + 1))

done

echo "The highest number is $max"

**6- código:  
#infelizmente não consegui entender a sintaxe de remover/deletar dentro de um comando (e não no terminal, diretamente); segue tentativa de programa**

Echo “Would you like to delete a file/directory? [y/n]”  
read ANSWER  
if [“ANSWER” = “y”]

then

echo “file or directory name:”

read “$ERASE”

if test –f”$ERASE”  
 then

rm –f $ERASE

elif test –d “ERASE”

then

rm –rf $ERASE

else

echo “invalid”

fi

elif [ “$ANSWER” = “n”]

then

exit

else

echo “invalid

fi