Name: Date:

(The recommended time is 30 minutes and the result should be at least 20 correct answers.)

LINUX/UNIX TEAM ENTRANCE TEST

1. **Print Linux kernel version.**
a) uname -r
b) lsb_release
c) uname -v
d) bootinfo -b

2. **Print the time of the last system boot.**
a) uptime
b) ntpdate
c) last
d) who -b

3. **Print the return code of the previous foreground process.**
a) echo $0
b) echo $$
c) echo $?
d) echo $!

4. **Execute the executable "script" file in the working directory in a subshell.**
a) . ~/.script
b) . ~/script
c) ./script
d) source ./script

5. **Execute the read-only "script" file in the working directory in the current shell.**
a) . ~/.script
b) sh script
c) ./script
d) source ./script

6. **Print the 25th line of the "access.log" file.**
a) cat -n 25 access.log
b) tail -25 access.log | head -1
c) head -25 access.log | tail -1
d) awk 'NR = 25' access.log

7. **List all directories (without files) in the working directory.**
a) ls -ld *
b) dir
c) ls -la
d) ls -l | grep '^d'

8. **Display all user names on the system.**
   a) users
   b) who
   c) awk '{print $1}' /etc/passwd
   d) egrep -o '^[^:]+' /etc/passwd

9. **Print all duplicate lines in the "test" file in the working directory.**
   a) uniq -d test
   b) uniq test | sort
   c) sort -u test
   d) sort test | uniq -d

10. **Remove the contents of the "notes" file in the working directory.**
   a) rm -f notes
   b) > notes
   c) cat notes | rm
   d) cat /dev/null >> notes

11. **Extract the "data.tar.gz" file in the working directory.**
   a) tar -xvzf data.tar.gz
   b) tar -xvjf data.tar.gz
   c) gunzip data.tar.gz | tar -xvf -
   d) gzip -d data.tar.gz | tar -xvf -

12. **Print the user name associated with the current effective user ID.**
   a) who am I
   b) whoami
   c) who -m
   d) logname

13. **Set the sticky bit on the "/usr/local/tmp" directory.**
   a) chmod 4777 /usr/local/tmp
   b) chmod 2777 /usr/local/tmp
   c) chmod +s /usr/local/tmp
   d) chmod +t /usr/local/tmp

14. **Set the ownership of the "/web" directory to the "apache" user.**
   a) chuser apache /web
   b) chuser :apache /web
   c) chown apache /web
   d) chown :apache /web

15. **Switch to the "root" user, including their environment settings.**
   a) su
   b) su root
   c) su -
   d) su -i

16. **Set default permissions for newly created files and directories so that the owner has all permissions, the group has read permissions and access to directories and others have no permissions.**
    a) umask 027
    b) umask 750
    c) umask 720
    d) umask 057

17. **Display a static list of all running processes with a detailed output (including the owner of the process, parent process identifier, CPU usage and the time the process started).**
    a) procinfo
    b) pstree
    c) proctree
    d) ps -ef

18. **Terminate the "27980" process immediately.**
    a) kill 27980
    b) kill -SIGTERM 27980
    c) kill -1 27980
    d) kill -SIGKILL 27980

19. **Display the status of all network interfaces.**
    a) ifconfig -all
    b) ifconfig -a
    c) ipconfig -all
    d) ipconfig -a

20. **Create users specified in the "users" file in the working directory and add them to the "admins" primary group.**
    a) while read `cat users`; do useradd -G admins $user; done
    b) while read `cat users`; do useradd -g admins $users; done
    c) for user in $(cat users); do useradd -g admins $user; done
    d) for user in $(cat users); do useradd -G admins $users; done

21. **Redirect the output of the "ls -l" command to the "log1" file in the working directory and display the number of lines on STDOUT.**
    a) ls -l > log1 | wc -l
    b) ls -l | wc -l | tee log1
    c) ls -l | tee log1 | wc -l
    d) wc -l $(ls -l) | tee log1

22. **If the working directory is the user's home directory, print "home", otherwise print the name of the working directory.**
    a) [$PWD = $HOME] && echo "home" || pwd
    b) [$PWD = $HOME] && echo "$HOME" || pwd
    c) [$(pwd) = home] && echo "$home" || pwd
    d) [$(pwd) = $home] && echo "home" || pwd

23. Redirect STDERR to the end of the "error.log" file in the working directory.
a) fsck /dev/sda1 2>&1 error.log
b) fsck /dev/sda1 2> | tee error.log
c) fsck /dev/sda1 2>> error.log
d) fsck /dev/sda1 2>>&1 | tee error.log

24. Display all open ports on the system including the protocol type and service name.
a) nmap -p - -sS -sU localhost | grep open
b) tcpdump -ni any | grep listen
c) netstat -tupln | grep open
d) ifstat -a | grep listen

25. Create the "-text" file in the working directory.
a) touch -- -text
b) touch -text
c) touch \-text
d) touch .-text

26. Execute the "sync" command in the background.
a) bg sync
b) sync &
c) nohup sync
d) sync | bg

27. Add the "scripts" directory from the user's home directory to the "PATH" variable.
a) PATH="$PATH:home/scripts"
b) PATH='$PATH:~$USER/scripts'
c) PATH='$PATH:$HOME/scripts'
d) PATH="$PATH:~/scripts"

28. Print lines starting with "Subject:" or "From:" of the "mail" file in the working directory.
a) grep '^(Subject|From):' mail
b) grep -E ^(Subject|From): mail
c) egrep "^(Subject|From):" mail
d) fgrep '^(Subject|From):' mail

29. Display disk space usage on all currently mounted file systems.
a) free
b) du
c) df
d) mount -a

30. Find all files with SUID permissions on the system.
a) find / -perm -4000
b) find / -perm 4000
c) find / -perm u=s
d) find / -perm u+s